ExtremeWare XOS Concepts Guide

Software Version 10.1

Extreme Networks, Inc.
3585 Monroe Street
Santa Clara, California 95051
(888) 257-3000
http://www.extremenetworks.com

Published: February 2004
Part number: 100150-00 Rev 02

©2004 Extreme Networks, Inc. All rights reserved. Extreme Networks, ExtremeWare and BlackDiamond are registered trademarks of Extreme Networks, Inc. in the United States and certain other jurisdictions. ExtremeWare XOS, ExtremeWare Vista, ExtremeWorks, ExtremeAssist, ExtremeAssist1, ExtremeAssist2, PartnerAssist, Extreme Standby Router Protocol, ESRP, SmartTraps, Alpine, Summit, Summit1, Summit4, Summit4/FX, Summit7i, Summit24, Summit48, Summit Virtual Chassis, SummitLink, SummitGbX, SummitRPS and the Extreme Networks logo are trademarks of Extreme Networks, Inc., which may be registered or pending registration in certain jurisdictions. The Extreme Turbodrive logo is a service mark of Extreme Networks, which may be registered or pending registration in certain jurisdictions. Specifications are subject to change without notice.

The ExtremeWare XOS operating system is based, in part, on the Linux operating system. The machine-readable copy of the corresponding source code is available for the cost of distribution. Please direct requests to Extreme Networks for more information at the following address:

Software Licensing Department
3585 Monroe Street
Santa Clara CA 95051

NetWare and Novell are registered trademarks of Novell, Inc. Merit is a registered trademark of Merit Network, Inc. Solaris and Java are trademarks of Sun Microsystems, Inc. F5, BIG/ip, and 3DNS are registered trademarks of F5 Networks, Inc. see/IT is a trademark of F5 Networks, Inc.

"Data Fellows", the triangle symbol, and Data Fellows product names and symbols/logos are trademarks of Data Fellows.

F-Secure SSH is a registered trademark of Data Fellows.

All other registered trademarks, trademarks and service marks are property of their respective owners.

3 4 5 6 7 8 9

Contents

Preface

Introduction **13**
Terminology 13

Conventions **14**

Related Publications **14**

Part 1 Using ExtremeWare XOS

Chapter 1 ExtremeWare XOS Overview

Virtual Routers 17
Virtual LANs (VLANs) 18
Spanning Tree Protocol 18
Quality of Service 18
Unicast Routing 18
IP Multicast Routing 19
Load Sharing 19

Chapter 2 Accessing the Switch

Understanding the Command Syntax **21**
Syntax Helper 22
Command Shortcuts 22
Modular Switch Numerical Ranges 23
Names 23
Symbols 24
Limits 24

Line-Editing Keys **24**

Command History **25**

Common Commands **25**

Configuring Management Access **27**

User Account 27
Administrator Account 27
Default Accounts 28
Creating a Management Account 29

Domain Name Service Client Services 29

Checking Basic Connectivity 30
Ping 30
Traceroute 30

Chapter 3 Managing the Switch

Overview 33

Understanding the XOS Shell 34
Configuring the Number of Active Shell Sessions 34

Using the Console Interface 34

Using the 10/100 Ethernet Management Port 34

Using Telnet 35
Connecting to Another Host Using Telnet 35
Configuring Switch IP Parameters 36
Disconnecting a Telnet Session 38

Using Trivial File Transfer Protocol (TFTP) 38
Connecting to Another Host Using TFTP 38
Enabling the TFTP Server 39

Using SNMP 39
Enabling and Disabling SNMPv1/v2c and SNMPv3 39
Accessing Switch Agents 40
Supported MIBs 40
Configuring SNMPv1/v2c Settings 40
Displaying SNMP Settings 40
SNMPv3 41
SNMPv3 Overview 41
Message Processing 42
SNMPv3 Security 42
MIB Access Control 44
Notification 45

Authenticating Users 48
RADIUS Client 48
TACACS+ 48
Configuring RADIUS Client and TACACS+ 48

Using the Simple Network Time Protocol 48
Configuring and Using SNTP 49
SNTP Example 52

Chapter 4 Configuring Slots and Ports on a Switch

Configuring a Slot on a Modular Switch **53**

Configuring Ports on a Switch **54**
Enabling and Disabling Switch Ports 54
Configuring Switch Port Speed and Duplex Setting 55

Jumbo Frames **56**
Enabling Jumbo Frames 56
Path MTU Discovery 56
IP Fragmentation with Jumbo Frames 57
IP Fragmentation within a VLAN 57

Load Sharing on the Switch **58**
Configuring Switch Load Sharing 58
Load-Sharing Examples 59
Verifying the Load-Sharing Configuration 59

Switch Port-Mirroring **59**
Modular Switch Port-Mirroring Example 60

Extreme Discovery Protocol **60**

Chapter 5 Virtual LANs (VLANs)

Overview of Virtual LANs **61**
Benefits 61

Types of VLANs **62**
Port-Based VLANs 62
Tagged VLANs 64
Protocol-Based VLANs 67
Precedence of Tagged Packets Over Protocol Filters 69

VLAN Names **70**
Default VLAN 70
Renaming a VLAN 70

Configuring VLANs on the Switch **71**
VLAN Configuration Examples 71

Displaying VLAN Settings **72**
Displaying Protocol Information 73

Chapter 6 Forwarding Database (FDB)

Overview of the FDB **75**
FDB Contents 75
How FDB Entries Get Added 75
FDB Entry Types 76
Disabling MAC Address Learning 77

FDB Configuration Examples **77**

MAC-Based Security **78**

Displaying FDB Entries **78**

Chapter 7 Quality of Service (QoS)

Overview of Policy-Based Quality of Service **82**

Applications and Types of QoS **82**
Voice Applications 82
Video Applications 83
Critical Database Applications 83
Web Browsing Applications 83
File Server Applications 83

Configuring QoS **84**

QoS Profiles **84**

Traffic Groupings **85**
Explicit Class of Service (802.1p and DiffServ) Traffic Groupings 86
Configuring DiffServ 87
Physical Groupings 89

Verifying Configuration and Performance **89**
QoS Monitor 90
Displaying QoS Profile Information 90

Chapter 8 Status Monitoring and Statistics

Status Monitoring **91**

Slot Diagnostics **91**
Running Diagnostics on I/O Modules 92
Running Diagnostics on MSM Modules 92
Viewing Slot Diagnostics 93

Port Statistics **93**

Port Errors **94**

Port Monitoring Display Keys **95**

System Temperature **95**

System Health Checking **96**

System Redundancy **96**
Electing the Node 97
Replicating Data Between the Nodes 98
Viewing Node Statistics 100

Event Management System/Logging **100**
Sending Event Messages to Log Targets 101

Filtering Events Sent to Targets 102
Formatting Event Messages 108
Displaying Real-Time Log Messages 109
Displaying Events Logs 109
Uploading Events Logs 109
Displaying Counts of Event Occurrences 110
Displaying Debug Information 111

Chapter 9 Security

Security Overview 113

Network Access Security 113

IP Access Lists (ACLs) 113
Creating IP Access Lists 114
ACL File Syntax 114
Example ACL Rule Entries 117
Using Access Lists on the Switch 118
Displaying and Clearing ACL Counters 119

Switch Protection 119
Policies 120
Creating Policies 120
Policy File Syntax 120
Policy Examples 125
Using Policies 129
Refreshing Policies 129
Saving Policies 129

Management Access Security 129

Authenticating Users Using RADIUS or TACACS+ 129
RADIUS 130
Configuring TACACS+ 132

Part 2 Using Switching and Routing Protocols

Chapter 10 Spanning Tree Protocol (STP)

Overview of the Spanning Tree Protocol 135
STP Terms 136

Spanning Tree Domains 137
Member VLANs 137
STPD Modes 138
Encapsulation Modes 139
STP States 140
Binding Ports 140
Rapid Root Failover 142

STP Configurations **142**
Basic STP Configuration 142
Multiple STPDs on a Port 145
VLAN Spanning Multiple STPDs 145
EMISTP Deployment Constraints 146

Per-VLAN Spanning Tree **148**
STPD VLAN Mapping 148
Native VLAN 148

Rapid Spanning Tree Protocol **148**
RSTP Terms 149
RSTP Concepts 149
RSTP Operation 152

STP Rules and Restrictions **159**

Configuring STP on the Switch **159**
STP Configuration Examples 160

Displaying STP Settings **163**

Chapter 11 Virtual Router Redundancy Protocol

Overview **165**

Determining the VRRP Master **166**
VRRP Tracking 166
Electing the Master Router 168

Additional VRRP Highlights **168**

VRRP Operation **169**
Simple VRRP Network Configuration 169
Fully-Redundant VRRP Network 170

VRRP Configuration Parameters **171**

VRRP Examples **172**
Configuring the Simple VRRP Network 172
Configuring the Fully-Redundant VRRP Network 173

Chapter 12 IP Unicast Routing

Overview of IP Unicast Routing **175**
Router Interfaces 176
Populating the Routing Table 177

Proxy ARP **178**
ARP-Incapable Devices 178
Proxy ARP Between Subnets 178

Relative Route Priorities **179**

Configuring IP Unicast Routing **179**

Verifying the IP Unicast Routing Configuration 180

Routing Configuration Example 180

Configuring DHCP/BOOTP Relay 182
Verifying the DHCP/BOOTP Relay Configuration 182
UDP Echo Server 183

Chapter 13 Interior Gateway Protocols

Overview 186
RIP Versus OSPF 186

Overview of RIP 187
Routing Table 187
Split Horizon 187
Poison Reverse 187
Triggered Updates 187
Route Advertisement of VLANs 187
RIP Version 1 Versus RIP Version 2 188

Overview of OSPF 188
Link-State Database 188
Areas 189
Point-to-Point Support 193

Route Re-Distribution 193
Configuring Route Re-Distribution 194
OSPF Timers and Authentication 195

RIP Configuration Example 196

Configuring OSPF 197
Configuring OSPF Wait Interval 197

OSPF Configuration Example 199
Configuration for ABR1 200
Configuration for IR1 200

Displaying OSPF Settings 200
OSPF LSDB Display 201

Chapter 14 Exterior Gateway Routing Protocols

Overview 204

BGP Attributes 204

BGP Communities 205

BGP Features 205
Route Reflectors 205
Route Confederations 207
Route Aggregation 211
Using the Loopback Interface 211

BGP Peer Groups 211
BGP Route Flap Dampening 212
BGP Route Selection 213
Stripping Out Private AS Numbers from Route Updates 214
Route Re-Distribution 214
BGP Static Network 215

Chapter 15 IP Multicast Routing

Overview 217
PIM Overview 218
IGMP Overview 219

Configuring IP Multicasting Routing 220

Configuration Examples 221
PIM-DM Configuration Example 221
Configuration for IR1 222
Configuration for ABR1 223

Part 3 Appendixes

Appendix A Software Upgrade and Boot Options

Downloading a New Image 227
Selecting a Primary or a Secondary Image 228
Understanding the Image Version String 228
Software Signatures 229
Rebooting the Switch 229

Saving Configuration Changes 229
Returning to Factory Defaults 230

Using TFTP to Upload the Configuration 231
Displaying Configuration Files 231
Renaming Configuration Files 231
Deleting Configuration Files 232

Using TFTP to Download the Configuration 232

Accessing the Bootloader 232

Appendix B Troubleshooting

LEDs 235

Using the Command Line Interface 236
Port Configuration 238
VLANs 238
STP 239

Debug Mode **240**

System Health Check **240**

System Odometer **240**

Contacting Extreme Technical Support **241**

Appendix C Supported Protocols, MIBs, and Standards

Index

Index of Commands

Preface

This preface provides an overview of this guide, describes guide conventions, and lists other publications that might be useful.

Introduction

This guide provides the required information to configure ExtremeWare XOS™ software running on either modular or stand-alone switches from Extreme Networks. The guide is intended for use by network administrators who are responsible for installing and setting up network equipment. It assumes a basic working knowledge of:

- Local area networks (LANs)
- Ethernet concepts
- Ethernet switching and bridging concepts
- Routing concepts
- Internet Protocol (IP) concepts
- Routing Information Protocol (RIP) and Open Shortest Path First (OSPF)
- Border Gateway Protocol (BGP-4) concepts
- IP Multicast concepts
- Protocol Independent Multicast (PIM) concepts
- Simple Network Management Protocol (SNMP)

NOTE

If the information in the release notes shipped with your switch differs from the information in this guide, follow the release notes.

Terminology

When features, functionality, or operation is specific to a modular or stand-alone switch family, the family name is used. Explanations about features and operations that are the same across all product families simply refer to the product as the “switch.”

Conventions

Table 1 and Table 2 list conventions that are used throughout this guide.

Table 1: Notice Icons

Icon	Notice Type	Alerts you to...
	Note	Important features or instructions.
	Caution	Risk of personal injury, system damage, or loss of data.
	Warning	Risk of severe personal injury.

Table 2: Text Conventions

Convention	Description
`Screen displays`	This typeface indicates command syntax, or represents information as it appears on the screen.
The words “enter” and “type”	When you see the word “enter” in this guide, you must type something, and then press the Return or Enter key. Do not press the Return or Enter key when an instruction simply says “type.”
[Key] names	Key names are written with brackets, such as [Return] or [Esc].
	If you must press two or more keys simultaneously, the key names are linked with a plus sign (+). Example:
	Press [Ctrl]+[Alt]+[Del].
Words in *italicized* type	Italics emphasize a point or denote new terms at the place where they are defined in the text.

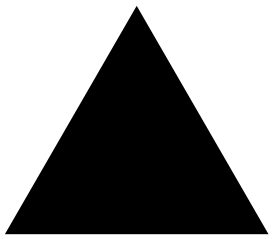
Related Publications

The publications related to this one are:

- ExtremeWare XOS release notes
- *ExtremeWare XOS 10.1 Command Reference Guide*
- *Extreme Networks BlackDiamond 10K-Series Switch Installation Guide*

Documentation for Extreme Networks products is available on the World Wide Web at the following location:

http://www.extremenetworks.com/



Part 1
Using ExtremeWare XOS

1 ExtremeWare XOS Overview

ExtremeWare XOS is the full-featured software operating system that is designed to run on the Extreme Networks BlackDiamond® 10800 family of switches.

NOTE

ExtremeWare XOS 10.1 only supports Extreme Networks BlackDiamond 10800 family products. This does not include other BlackDiamond families, Alpine®, and Summit™ "i", Summit 24e3, and Summit 200 series platforms.

Virtual Routers

ExtremeWare XOS 10.1 introduces support for virtual routers. This capability allows a single physical switch to be split into multiple virtual routers. This feature separates the traffic forwarded by a virtual router from the traffic on a different virtual router. Each virtual router maintains a separate logical forwarding table, which allows the virtual routers to have overlapping address spaces. Since each virtual router maintains its own separate routing information, and switch ports can belong to one and only one virtual router, packets arriving at a port on one virtual router can never be switched to the ports on another. In this release of ExtremeWare XOS, the management port belongs to one virtual router and all other ports belong to another virtual router.

With multiple virtual routers contained on a single physical switch, some commands in ExtremeWare XOS now require you to specify to which virtual router the command applies. For example, when you use the ping command, you must specify from which virtual router the ping packets are generated. Many commands that deal with switch management use the management virtual router by default. See the *ExtremeWare XOS Command Reference Guide* for information on the defaults for individual commands.

NOTE

The term "Virtual Router" is also used with the Virtual Router Redundancy Protocol (VRRP). VRRP uses the term to refer to a single virtual router that spans more than one physical router, which allows multiple switches to provide redundant routing services to users. For more information about VRRP, see Chapter 11.

Virtual LANs (VLANs)

ExtremeWare XOS has a VLAN feature that enables you to construct your broadcast domains without being restricted by physical connections. A VLAN is a group of location- and topology-independent devices that communicate as if they were on the same physical local area network (LAN).

Implementing VLANs on your network has the following three advantages:

- Help to control broadcast traffic. If a device in VLAN *Marketing* transmits a broadcast frame, only VLAN *Marketing* devices receive the frame.
- Provide extra security. Devices in VLAN *Marketing* can only communicate with devices on VLAN *Sales* using routing services.
- Ease the change and movement of devices on networks.

NOTE

For more information on VLANs, see Chapter 5.

Spanning Tree Protocol

The switch supports the IEEE 802.1D Spanning Tree Protocol (STP), which is a bridge-based mechanism for providing fault tolerance on networks. STP enables you to implement parallel paths for network traffic, and ensure that redundant paths are:

- Disabled when the main paths are operational.
- Enabled if the main traffic paths fail.

A single spanning tree can span multiple VLANs.

NOTE

For more information on STP, see Chapter 10.

Quality of Service

ExtremeWare XOS has Policy-Based Quality of Service (QoS) features that enable you to specify service levels for different traffic groups. By default, all traffic is assigned the *normal* QoS policy profile. If needed, you can customize other QoS policies and apply them to different traffic types so that they have different guaranteed minimum bandwidth, maximum bandwidth, and priority.

NOTE

For more information on Quality of Service, see Chapter 7.

Unicast Routing

The switch can route IP traffic between the VLANs that are configured as virtual router interfaces. Both dynamic and static IP routes are maintained in the routing table. The following routing protocols are supported:

- RIP version 1
- RIP version 2

- OSPF version 2
- BGP version 4

NOTE

For more information on IP unicast routing, see Chapter 12. For more information on RIP, see Chapter 20.

IP Multicast Routing

The switch can use IP multicasting to allow a single IP host to transmit a packet to a group of IP hosts. ExtremeWare XOS supports multicast routes that are learned by the Protocol Independent Multicast (dense mode or sparse mode).

NOTE

For more information on IP multicast routing, see Chapter 15.

Load Sharing

Load sharing allows you to increase bandwidth and resiliency by using a group of ports to carry traffic in parallel between systems. The load sharing algorithm allows the switch to use multiple ports as a single logical port. For example, VLANs see the load-sharing group as a single virtual port. The algorithm also guarantees packet sequencing between clients.

NOTE

For information on load sharing, see Chapter 4.

2 Accessing the Switch

This chapter covers the following topics:

- Understanding the Command Syntax on page 21
- Line-Editing Keys on page 24
- Command History on page 25
- Common Commands on page 25
- Configuring Management Access on page 27
- Domain Name Service Client Services on page 29
- Checking Basic Connectivity on page 30

Understanding the Command Syntax

This section describes the steps to take when entering a command. Refer to the sections that follow for detailed information on using the command line interface (CLI).

ExtremeWare XOS command syntax is described in detail in the *ExtremeWare XOS Command Reference Guide*. Some commands are also described in this user guide, in order to describe how to use the features of the ExtremeWare XOS software. However, only a subset of commands are described here, and in some cases only a subset of the options that a command supports. The *ExtremeWare XOS Command Reference Guide* should be considered the definitive source for information on ExtremeWare XOS commands.

You may only enter configuration commands at the # prompt. As you are booting up, you may see the > command prompt. At the > prompt, you may only enter monitoring commands, not configuration commands. When the bootup process is complete, the # prompt appears.

When entering a command at the prompt, ensure that you have the appropriate privilege level. Most configuration commands require you to have the administrator privilege level. To use the command line interface (CLI), follow these steps:

1 Enter the command name.

If the command does not include a parameter or values, skip to step 3. If the command requires more information, continue to step 2.

2 If the command includes a parameter, enter the parameter name and values.

3 The value part of the command specifies how you want the parameter to be set. Values include numerics, strings, or addresses, depending on the parameter.

4 After entering the complete command, press [Return].

NOTE

If an asterisk (*) appears in front of the command line prompt, it indicates that you have outstanding configuration changes that have not been saved. For more information on saving configuration changes, see Appendix A.

Syntax Helper

The CLI has a built-in syntax helper. If you are unsure of the complete syntax for a particular command, enter as much of the command as possible and press [Tab] or [?]. The syntax helper provides a list of options for the remainder of the command, and places the cursor at the end of the command you have entered so far, ready for the next option.

If you enter an invalid commend, the syntax helper notifies you of your error and places the cursor at the place where the error seems to have occurred.

If the command is one where the next option is a named component, such as a VLAN, access profile, or route map, the syntax helper will also list any currently configured names that might be used as the next option. In situations where this list might be very long, the syntax helper will list only one line of names, followed by an ellipses to indicate that there are more names that can be displayed.

The syntax helper also provides assistance if you have entered an incorrect command.

Abbreviated Syntax

Abbreviated syntax is the shortest unambiguous allowable abbreviation of a command or parameter. Typically, this is the first three letters of the command. If you do not enter enough letters to allow the switch to determine which command you mean, the syntax helper will provide a list of the options based on the portion of the command you have entered.

NOTE

When using abbreviated syntax, you must enter enough characters to make the command unambiguous and distinguishable to the switch.

Command Shortcuts

Components are typically named using the `create` command. When you enter a command to configure a named component, you do not need to use the keyword of the component. For example, to create a VLAN, enter a VLAN name:

```
create vlan engineering
```

After you have created the name for the VLAN, you can then eliminate the keyword `vlan` from all other commands that require the name to be entered. For example, instead of entering the modular switch command

```
configure vlan engineering delete port 1:3,4:6
```

you could enter the following shortcut:

```
configure engineering delete port 1:3,4:6
```

Similarly, on the stand-alone switch, instead of entering the command

```
configure vlan engineering delete port 1-3,6
```

you could enter the following shortcut:

```
configure engineering delete port 1-3,6
```

Although it is helpful to have unique names for system components, this is not a requirement. If ExtremeWare XOS encounters any ambiguity in the components within your command, it generates a message requesting that you clarify the object you specified.

Modular Switch Numerical Ranges

Commands that require you to enter one or more port numbers on a modular switch use the parameter <portlist> in the syntax. A <portlist> can be one port on a particular slot. For example,

```
port 3:1
```

A <portlist> can be a range of numbers. For example,

```
port 3:1-3:3
```

You can add additional slot and port numbers to the list, separated by a comma:

```
port 3:1,4:8,6:10
```

You can specify all ports on a particular slot. For example,

```
port 3:*
```

indicates all ports on slot 3.

You can specify a range of slots and ports. For example,

```
port 2:3-4:5
```

indicates slot 2, port 3 through slot 4, port 5.

Names

All named components within a category of the switch configuration, such as VLAN, must have a unique name. Names can be re-used across categories, however. Names must begin with an alphabetical character and cannot contain any spaces. The maximum length for a name is 32 characters. Names may contain alphanumeric characters and underscores (_) and cannot be keywords, such as vlan, stp, and so on.

Symbols

You may see a variety of symbols shown as part of the command syntax. These symbols explain how to enter the command, and you do not type them as part of the command itself. Table 3 summarizes command syntax symbols.

Table 3: Command Syntax Symbols

Symbol	Description
angle brackets < >	Enclose a variable or value. You must specify the variable or value. For example, in the syntax `configure vlan <vlan name> ipaddress <ipaddress>` you must supply a VLAN name for `<vlan name>` and an address for `<ip_address>` when entering the command. Do not type the angle brackets.
square brackets [ ]	Enclose a required value or list of required arguments. One or more values or arguments can be specified. For example, in the syntax `use image [primary | secondary]` you must specify either the primary or secondary image when entering the command. Do not type the square brackets.
vertical bar |	Separates mutually exclusive items in a list, one of which must be entered. For example, in the syntax `configure snmp community [read-only | read-write] <string>` you must specify either the read or write community string in the command. Do not type the vertical bar.
braces { }	Enclose an optional value or a list of optional arguments. One or more values or arguments can be specified. For example, in the syntax `reboot {<date> <time> | cancel}` you can specify either a particular date and time combination, or the keyword `cancel` to cancel a previously scheduled reboot. If you do not specify an argument, the command will prompt, asking if you want to reboot the switch now. Do not type the braces.

Limits

The command line can process up to 512 characters, including spaces. If you attempt to enter more than 512 characters, the switch emits an audible “beep” and will not accept any further input. The first 512 characters are processed, however.

Line-Editing Keys

Table 4 describes the line-editing keys available using the CLI.

Table 4: Line-Editing Keys

Key(s)	Description
Left arrow or [Ctrl] + B	Moves the cursor one character to the left.
Right arrow or [Ctrl] + F	Moves the cursor one character to the right.

Table 4: Line-Editing Keys (continued)

Key(s)	Description
[Ctrl] + H or Backspace	Deletes character to left of cursor and shifts remainder of line to left.
Delete or [Ctrl] + D	Deletes character under cursor and shifts remainder of line to left.
[Ctrl] + K	Deletes characters from under cursor to end of line.
Insert	Toggles on and off. When toggled on, inserts text and shifts previous text to right.
Left Arrow	Moves cursor to left.
Right Arrow	Moves cursor to right.
Home or [Ctrl] + A	Moves cursor to first character in line.
End or [Ctrl] + E	Moves cursor to last character in line.
[Ctrl] + L	Clears screen and movers cursor to beginning of line.
[Ctrl] + P or Up Arrow	Displays previous command in command history buffer and places cursor at end of command.
[Ctrl] + N or Down Arrow	Displays next command in command history buffer and places cursor at end of command.
[Ctrl] + U	Clears all characters typed from cursor to beginning of line.
[Ctrl] + W	Deletes previous word.
[Ctrl] + C	Interrupts the current CLI command execution.

Command History

ExtremeWare XOS “remembers” all the commands you enter. You can display a list of these commands by using the following command:

```
history
```

Common Commands

Table 5 describes some of the common commands used to manage the switch. Commands specific to a particular feature may also be described in other chapters of this guide. For a detailed description of the commands and their options, see the *ExtremeWare XOS Command Reference Guide.*

Table 5: Common Commands

Command	Description
clear session <sessId> | all	Terminates a Telnet session from the switch.
configure account <name> {password}	Configures a user account password. The switch will interactively prompt for a new password, and for reentry of the password to verify it. Passwords must have a minimum of 1 character and can have a maximum of 30 characters. Passwords are case-sensitive; user names are not case sensitive.

Table 5: Common Commands (continued)

Command	Description
configure banner	Configures the banner string. You can enter up to 24 rows of 79-column text that is displayed before the login prompt of each session. Press [Return] at the beginning of a line to terminate the command and apply the banner. To clear the banner, press [Return] at the beginning of the first line.
configure ports <port_list> auto off {speed [10 | 100 | 1000]} duplex [half | full]	Manually configures the port speed and duplex setting of one or more ports on a switch.
configure slot <slot> module <module_type>	Configures a slot for a particular I/O module card.
configure time <month> <day> <year> <hour> <min> <sec>	Configures the system date and time. The format is as follows: `mm/dd/yyyy hh:mm:ss` The time uses a 24-hour clock format. You cannot set the year past 2036.
configure timezone {name <tz_name>} <GMT_offset> {autodst {name <dst_timezone_ID>} {<dst_offset>} {begins [every <floatingday> | on <absoluteday>] {at <time_of_day>} {ends [every <floatingday> | on <absoluteday>] {at <time_of_day>}}} | noautodst}	Configures the time zone information to the configured offset from GMT time. The format of `gmt_offset` is +/- minutes from GMT time. The `autodst` and `noautodst` options enable and disable automatic Daylight Saving Time change based on the North American standard. Additional options are described in the *ExtremeWare XOS Command Reference Guide*.
configure vlan <vlan_name> ipaddress <ipaddress> {<ipNetmask>}	Configures an IP address and subnet mask for a VLAN.
create account [admin | user] <account-name> {<password>}	Creates a user account. This command is available to admin-level users and to users with RADIUS command authorization. The username is between 1 and 30 characters, the password is between 0 and 30 characters.
create vlan <vlan_name>	Creates a VLAN.
delete account <name>	Deletes a user account.
delete vlan <vlan_name>	Deletes a VLAN.
disable bootp vlan [<vlan> | all]	Disables BOOTP for one or more VLANs.
disable clipaging	Disables pausing of the screen display when a show command output reaches the end of the page.
disable idletimeout	Disables the timer that disconnects all sessions. After being disabled, console sessions remain open until the switch is rebooted or you logoff. Telnet sessions remain open until you close the Telnet client.
disable port [<port_list> | all]	Disables a port on the switch.
disable telnet	Disables Telnet access to the switch.
enable bootp vlan [<vlan> | all]	Enables BOOTP for one or more VLANs.
enable clipaging	Enables pausing of the screen display when `show command` output reaches the end of the page. The default setting is enabled.
enable idletimeout	Enables a timer that disconnects all sessions (both Telnet and console) after 20 minutes of inactivity. The default setting is disabled.
history	Displays the previous 49 commands entered on the switch.
show banner	Displays the user-configured banner.

Table 5: Common Commands (continued)

Command	Description
unconfigure switch {all}	Resets all switch parameters (with the exception of defined user accounts, and date and time information) to the factory defaults. If you specify the keyword `all`, the switch erases the currently selected configuration image in flash memory and reboots. As a result, all parameters are reset to default settings.

Configuring Management Access

ExtremeWare XOS supports the following two levels of management:

- User
- Administrator

In addition to the management levels, you can optionally use an external RADIUS server to provide CLI command authorization checking for each command. For more information on RADIUS, see "RADIUS Client" in Chapter 3.

User Account

A user-level account has viewing access to all manageable parameters, with the exception of:

- User account database.
- SNMP community strings.

A user-level account can use the `ping` command to test device reachability, and change the password assigned to the account name. If you have logged on with user capabilities, the command line prompt ends with a (>) sign. For example:

```
BD-1.2 >
```

Administrator Account

An administrator-level account can view and change all switch parameters. It can also add and delete users, and change the password associated with any account name. The administrator can disconnect a management session that has been established by way of a Telnet connection. If this happens, the user logged on by way of the Telnet connection is notified that the session has been terminated.

If you have logged on with administrator capabilities, the command line prompt ends with a (#) sign. For example:

```
BD-1.18 #
```

Prompt Text

The prompt text is taken from the SNMP `sysname` setting. The number that follows the colon indicates the sequential line/command number.

If an asterisk (*) appears in front of the command line prompt, it indicates that you have outstanding configuration changes that have not been saved. For example:

```
* BD-1.19 #
```

Default Accounts

By default, the switch is configured with two accounts, as shown in Table 6.

Table 6: Default Accounts

Account Name	Access Level
admin	This user can access and change all manageable parameters. However, the user may not delete all admin accounts.
user	This user can view (but not change) all manageable parameters, with the following exceptions: • This user cannot view the user account database. • This user cannot view the SNMP community strings.

Changing the Default Password

Default accounts do not have passwords assigned to them. Passwords can have a minimum of zero characters and can have a maximum of 30 characters.

NOTE

Passwords are case-sensitive; user names are not case-sensitive.

To add a password to the default admin account:

1. Log in to the switch using the name *admin*.
2. At the password prompt, press [Return].
3. Add a default admin password by entering the following command:

   ```
   configure account admin
   ```

4. Enter the new password at the prompt.
5. Re-enter the new password at the prompt.

To add a password to the default user account:

1. Log in to the switch using the name *admin*.
2. At the password prompt, press [Return], or enter the password that you have configured for the *admin* account.
3. Add a default user password by entering the following command:

   ```
   configure account user
   ```

4. Enter the new password at the prompt.
5. Re-enter the new password at the prompt.

NOTE

If you forget your password while logged out of the command line interface, contact your local technical support representative, who will advise on your next course of action.

Creating a Management Account

The switch can have a total of 16 management accounts. You can use the default names (*admin* and *user*), or you can create new names and passwords for the accounts. Passwords can have a minimum of 0 characters and can have a maximum of 30 characters.

To create a new account:

1. Log in to the switch as *admin*.
2. Add a new user by using the following command:

   ```
   create account [admin | user] <username> {<password>}
   ```

Viewing Accounts

To view the accounts that have been created, you must have administrator privileges. To see the accounts, use the following command:

```
show accounts
```

Deleting an Account

To delete a account, you must have administrator privileges. To delete an account, use the following command:

```
delete account <name>
```

Domain Name Service Client Services

The Domain Name Service (DNS) client in ExtremeWare XOS augments the following commands to allow them to accept either IP addresses or host names:

- `telnet`
- `download [bootrom | configuration | image]`
- `ping`
- `traceroute`

In addition, the `nslookup` utility can be used to return the IP address of a hostname.

You can specify up to eight DNS servers for use by the DNS client using the following command:

```
configure dns-client add domain-suffix <domain_name> | name-server <ip_address>
```

You can specify a default domain for use when a host name is used without a domain. Use the following command:

```
configure dns-client default-domain <domain_name>
```

For example, if you specify the domain “xyz-inc.com” as the default domain, then a command such as `ping accounting1` will be taken as if it had been entered `ping accounting1.xyz-inc.com`.

Checking Basic Connectivity

The switch offers the following commands for checking basic connectivity:

- `ping`
- `traceroute`

Ping

The `ping` command enables you to send Internet Control Message Protocol (ICMP) echo messages to a remote IP device. The `ping` command is available for both the user and administrator privilege level.

The `ping` command syntax is:

```
ping {udp} {[continuous | count <count>]} {start-size <start-size>} {end-size
<end-size}} {interval <interval>} {ttl <ttl>} {tos <tos>} {vr <vrid>} <host>
```

Options for the ping command are described in Table 7.

Table 7: Ping Command Parameters

Parameter	Description
udp	Specifies that the ping request should use UDP instead of ICMP.
continuous	Specifies that UDP or ICMP echo messages to be sent continuously. This option can be interrupted by pressing any key.
start-size	Specifies the size, in bytes, of the packet to be sent, or the starting size if incremental packets are to be sent.
end-size	Specifies the end size, in bytes, of the packet to be sent, when incremental packets are sent.
vr	Specifies the virtual route to use for sending out the echo message. If not specified, the virtual router assigned to the *default* VLAN is used.
tos	Sets the TOS value.
ttl	Sets the TTL value.
dont-fragment	Sets the IP to not fragment the bit.
interval	Sets the time interval between sending out ping requests.
host	Specifies a IPV4 host to ping.

If a `ping` request fails, the switch continues to send `ping` messages until interrupted. Press [Control] + C to interrupt a `ping` request. The statistics are tabulated after the ping is interrupted.

Traceroute

The `traceroute` command enables you to trace the routed path between the switch and a destination endstation. The `traceroute` command syntax is:

```
traceroute {vrid <vrid>} <host> {from <source IP address>} {ttl <number>} {port <port
number>}
```

where:

- `ip_address` is the IP address of the destination endstation.
- `hostname` is the hostname of the destination endstation. To use the hostname, you must first configure DNS.
- `from` uses the specified source address in the ICMP packet. If not specified, the address of the transmitting interface is used.
- `ttl` configures the switch to trace the hops until the time-to-live has been exceeded for the switch.
- `port` uses the specified UDP port number.

3 Managing the Switch

This chapter covers the following topics:

- Overview on page 33
- Understanding the XOS Shell on page 34
- Using the Console Interface on page 34
- Using the 10/100 Ethernet Management Port on page 34
- Using Telnet on page 35
- Using Trivial File Transfer Protocol (TFTP) on page 38
- Using SNMP on page 39
- Authenticating Users on page 48
- Using the Simple Network Time Protocol on page 48

Overview

Using ExtremeWare XOS, you can manage the switch using the following methods:

- Access the CLI by connecting a terminal (or workstation with terminal-emulation software) to the console port.
- Access the switch remotely using TCP/IP through one of the switch ports or through the dedicated 10/100 unshielded twisted pair (UTP) Ethernet management port. Remote access includes:
  - Telnet using the CLI interface.
  - SNMP access using EPICenter or another SNMP manager.
- Download software updates and upgrades. For more information, see Appendix A, Software Upgrade and Boot Options.

The switch supports up to the following number of concurrent user sessions:

- One console session
  - Two console sessions are available if two management modules are installed.
- Eight shell sessions
- Eight Telnet sessions
- Eight TFTP sessions

Understanding the XOS Shell

When you login to ExtremeWare XOS from a terminal, you enter the XOS shell and the XOS shell prompt is displayed. At the shell prompt, you input the commands to be executed on the switch. After the switch processes and executes a command, the results are relayed to and displayed on your terminal.

The XOS shell supports ANSI, VT100, and XTERM terminal emulation, and the shell adjusts to the correct terminal type and window size. In addition, the XOS shell supports UNIX style pageview for page-by-page command output capability.

Up to eight active shell sessions can access the switch concurrently.

For more information about the line-editing keys that you can use with the XOS shell, see “Line-Editing Keys” on page 24.

Configuring the Number of Active Shell Sessions

You can change the number of simultaneous, active XOS shell sessions supported by the switch. By default, eight active shell sessions can access the switch concurrently. If you configure a new limit, only new incoming XOS shell sessions are affected. If you decrease the limit and the current number of sessions already exceeds the new maximum, the switch refuses only new incoming connections until the number of shell session drops below the new limit. Already connected shell sessions will not be disconnected as a result of decreasing the new limit.

To configure the amount of XOS shell sessions, use the following command:

```
configure cli max-sessions
```

Using the Console Interface

The CLI built into the switch is accessible by way of the 9-pin, RS-232 port labeled *console*, located on the front of the modular switch management module.

NOTE

For more information on the console port pinouts, see the hardware installation guide that shipped with your switch.

After the connection has been established, you will see the switch prompt and you can log in.

Using the 10/100 Ethernet Management Port

The MSM provides a dedicated 10/100 Ethernet management port. This port provides dedicated remote access to the switch using TCP/IP. It supports the following management methods:

- Telnet using the CLI interface
- SNMP access using EPICenter or another SNMP manager

The management port on the MSM is a DTE port. The TCP/IP configuration for the management port is done using the same syntax as used for VLAN configuration. The VLAN *mgmt* comes preconfigured with only the 10/100 UTP management port as a member.

When you configure the IP address for the VLAN *mgmt*, it gets assigned to the primary MSM. You can connect to the management port on the primary MSM for any switch configuration. The management port on the backup MSM is available only when failover occurs. At that time, the primary MSM relinquishes its role, the backup MSM takes over, and the VLAN *mgmt* on the new primary MSM acquires the ipaddress of the previous master.

You can configure the IP address, subnet mask, and default router for the VLAN *mgmt*, using the following commands:

```
configure vlan mgmt ipaddress <ip_address>/<subnet_mask>
configure iproute add default <gateway> {vr <vrname>} {<metric>} {multicast-only |
unicast-only}
```

Using Telnet

Any workstation with a Telnet facility should be able to communicate with the switch over a TCP/IP network using VT100 terminal emulation.

Up to eight active Telnet sessions can access the switch concurrently. If `idletimeouts` are enabled, the Telnet connection will time out after 20 minutes of inactivity. If a connection to a Telnet session is lost inadvertently, the switch terminates the session within two hours.

Before you can start a Telnet session, you must set up the IP parameters described in “Configuring Switch IP Parameters” on page 36. Telnet is enabled by default.

NOTE

Maximize the Telnet screen so that automatically updating screens display correctly.

To open the Telnet session, you must specify the IP address of the device that you want to manage. Check the user manual supplied with the Telnet facility if you are unsure of how to do this.

After the connection is established, you will see the switch prompt and you may log in.

Connecting to Another Host Using Telnet

You can Telnet from the current CLI session to another host using the following command:

```
telnet [<remote_ip> | <host_name>] {vr <vr_name>} {<port>}
```

If the TCP port number is not specified, the Telnet session defaults to port 23. If the virtual router name is not specified, the Telnet session defaults to virtual router VR-0. Only VT100 emulation is supported.

To change the default TCP port number, use the following command:

```
configure telnet port [<port number> | default]
```

The range for the port number is 1 through 65535.

Configuring Switch IP Parameters

To manage the switch by way of a Telnet connection or by using an SNMP Network Manager, you must first configure the switch IP parameters.

Using a BOOTP or DHCP Server

If you are using IP and you have a Bootstrap Protocol (BOOTP) server set up correctly on your network, you must provide the following information to the BOOTP server:

- Switch Media Access Control (MAC) address, found on the rear label of the switch
- IP address
- Subnet address mask (optional)

After this is done, the IP address and subnet mask for the switch will be downloaded automatically. You can then start managing the switch using this addressing information without further configuration. If you use a DHCP server, make sure it is enabled on the required VLAN to receive IP configuration information.

You can enable BOOTP or DHCP on a per-VLAN basis by using the following commands:

```
enable bootp vlan [<vlan> | all]
enable dhcp vlan [<vlan_name> | all]
```

You can disable BOOTP or DHCP on a per-VLAN basis by using the following commands:

```
disable bootp vlan [<vlan> | all]
disable dhcp vlan [<vlan_name> | all]
```

To view the current state of the BOOTP or DHCP client, use the following command:

```
show dhcp-client state
```

If you configure the switch to use BOOTP, the switch IP address is not retained through a power cycle, even if the configuration has been saved. To retain the IP address through a power cycle, you must configure the IP address of the VLAN using the command-line interface or Telnet.

All VLANs within a switch that are configured to use BOOTP to get their IP address use the same MAC address. Therefore, if you are using BOOTP relay through a router, the BOOTP server relays packets based on the gateway portion of the BOOTP packet.

NOTE

For more information on DHCP/BOOTP relay, see Chapter 12.

Manually Configuring the IP Settings

If you are using IP without a BOOTP server, you must enter the IP parameters for the switch in order for the SNMP Network Manager or Telnet software to communicate with the device. To assign IP parameters to the switch, you must perform the following tasks:

- Log in to the switch with administrator privileges using the console interface.
- Assign an IP address and subnet mask to a VLAN.

The switch comes configured with a default VLAN named *default*. To use Telnet or an SNMP Network Manager, you must have at least one VLAN on the switch, and it must be assigned an IP address and subnet mask. IP addresses are always assigned to each VLAN. The switch can be assigned multiple IP addresses.

NOTE

For information on creating and configuring VLANs, see Chapter 5.

To manually configure the IP settings:

1. Connect a terminal or workstation running terminal-emulation software to the console port, as detailed in "Using the Console Interface" on page 34.
2. At your terminal, press [Return] one or more times until you see the login prompt.
3. At the login prompt, enter your user name and password. Note that they are both case-sensitive. Ensure that you have entered a user name and password with administrator privileges.
   - If you are logging in for the first time, use the default user name *admin* to log in with administrator privileges. For example:

     ```
     login: admin
     ```

     Administrator capabilities enable you to access all switch functions. The default user names have no passwords assigned.
   - If you have been assigned a user name and password with administrator privileges, enter them at the login prompt.
4. At the password prompt, enter the password and press [Return].

   When you have successfully logged in to the switch, the command-line prompt displays the name of the switch in its prompt.
5. Assign an IP address and subnetwork mask for the default VLAN by using the following command:

   ```
   configure vlan <vlan_name> ipaddress <ipaddress> {<ipNetmask>}
   ```

   For example:

   ```
   configure vlan default ipaddress 123.45.67.8 255.255.255.0
   ```

   Your changes take effect immediately.

   NOTE

   As a general rule, when configuring any IP addresses for the switch, you can express a subnet mask by using dotted decimal notation, or by using classless inter-domain routing notation (CIDR). CIDR uses a forward slash plus the number of bits in the subnet mask. Using CIDR notation, the command identical to the one above would be:

   ```
   configure vlan default ipaddress 123.45.67.8 / 24
   ```
6. Configure the default route for the switch using the following command:

   ```
   configure iproute add default <gateway> {vr <vrname>} {<metric>} {multicast-only |
   unicast-only}
   ```

   For example:

   ```
   configure iproute add default 123.45.67.1
   ```

7 Save your configuration changes so that they will be in effect after the next switch reboot, by typing:

```
save
```

8 When you are finished using the facility, log out of the switch by typing:

```
logout
```
or
```
quit
```

Disconnecting a Telnet Session

An administrator-level account can disconnect a Telnet management session. If this happens, the user logged in by way of the Telnet connection is notified that the session has been terminated.

To terminate a Telnet session:

1 Log in to the switch with administrator privileges.

2 Determine the session number of the session you want to terminate by using the following command:

```
show session
```

3 Terminate the session by using the following command:

```
clear session <sessId> | all
```

Using Trivial File Transfer Protocol (TFTP)

ExtremeWare XOS supports the client portion of the Trivial File Transfer Protocol (TFTP) based on RFC 1350. TFTP is a method used to transfer files from one network device to another. The ExtremeWare XOS TFTP client is a command line application used to contact an external TFTP server on the network. For example, XOS uses TFTP to download software image files, switch configuration files, and access control lists (ACLs) from a server on the network to the switch.

Up to eight active TFTP sessions can run on the switch concurrently.

For detailed information about downloading software image files, see Chapter A, “Software Upgrade and Boot Options.”

For detailed information about downloading ACLs, see Chapter 9, “Security.”

Connecting to Another Host Using TFTP

You can TFTP from the current CLI session to another host using the following command:

```
tftp [<ip_address> | <host_name>] {-v <vr_id>} [-g | -p] [{-l <local_file>} {-r
<remote_file>} | {-r <remote_file>} {-l <local_file>}]
```

The TFTP session defaults to port 69.

For example, to connect to a remote TFTP server and “get” or retrieve an ExtremeWare XOS image file from that host, use the following command:

```
tftp 10.123.45.67 -g -r bd10ki386-10.1.086.tgz
```

When you “get” the file via TFTP, the switch saves the file to the primary MSM. If the switch detects a backup MSM in the running state, the file is replicated to the backup MSM.

To view the files you retrieved, enter the `ls` command at the command prompt.

Enabling the TFTP Server

By default, the TFTP server is disabled on the switch. You can choose to enable the TFTP server by using the following command:

```
enable tftp
```

To disable the TFTP server on the switch, use the following command

```
disable tftp
```

You must be logged in as an administrator to enable or disable the TFTP server.

To change the default TCP server/daemon port number, use the following command:

```
configure tftp port [<portno> | default]
```

The range for the port number is 1 through 65535.

Using SNMP

Any Network Manager running the Simple Network Management Protocol (SNMP) can manage the switch, provided the Management Information Base (MIB) is installed correctly on the management station. Each Network Manager provides its own user interface to the management facilities.

The following sections describe how to get started if you want to use an SNMP manager. It assumes you are already familiar with SNMP management. If not, refer to the following publication:

> *The Simple Book*
> by Marshall T. Rose
> ISBN 0-13-8121611-9
> Published by Prentice Hall.

The ExtremeWare XOS SNMP agent supports the same MIBs and is backwards compatible with the ExtremeWare SNMP agent.

Enabling and Disabling SNMPv1/v2c and SNMPv3

ExtremeWare XOS can concurrently support SNMPv1/v2c and SNMPv3. The default for the switch is to have both types of SNMP enabled. Network managers can access the device with either SNMPv1/v2c methods or SNMPv3. To enable concurrent support, use the following command:

```
enable snmp access
```

To prevent any type of SNMP access, use the following command:

```
disable snmp access
```

To prevent access using SNMPv1/v2c methods and allow access using SNMPv3 methods only, use the following commands:

```
enable snmp access
disable snmp access {snmp-v1v2c}
```

There is no way to configure the switch to allow SNMPv1/v2c access and prevent SNMPv3 access.

Most of the commands that support SNMPv1/v2c use the keyword `snmp`; most of the commands that support SNMPv3 use the keyword `snmpv3`.

Accessing Switch Agents

To have access to the SNMP agent residing in the switch, at least one VLAN must have an IP address assigned to it.

By default, SNMP access and SNMPv1/v2c traps are enabled. SNMP access and SNMP traps can be disabled and enabled independently—you can disable SNMP access but still allow SNMP traps to be sent, or vice versa.

Supported MIBs

In addition to private MIBs, the switch supports the standard MIBs listed in Appendix C.

Configuring SNMPv1/v2c Settings

The following SNMPv1/v2c parameters can be configured on the switch:

- **Authorized trap receivers**—An authorized trap receiver can be one or more network management stations on your network. The switch sends SNMPv1/v2c traps to all trap receivers. You can have a maximum of 16 trap receivers configured for each switch, and you can specify a community string and UDP port for individually for each trap receiver. All community strings must also be added to the switch using the `configure snmp add community` command.

  To configure a trap receiver on a switch, use the following command:

  ```
  configure snmp add trapreceiver <ip_address> community {hex} <community_name>
  {port <port_number>}
  ```

  You can delete a trap receiver using the `configure snmp delete trapreceiver` command.

- **Community strings**—The community strings allow a simple method of authentication between the switch and the remote Network Manager. There are two types of community strings on the switch. Read community strings provide read-only access to the switch. The default read-only community string is *public*. Read-write community strings provide read and write access to the switch. The default read-write community string is *private*.
- **System contact** (optional)—The system contact is a text field that enables you to enter the name of the person(s) responsible for managing the switch.
- **System name**—The system name is the name that you have assigned to this switch. The default name is the model name of the switch (for example, BD-1.2).
- **System location** (optional)—Using the system location field, you can enter an optional location for this switch.

Displaying SNMP Settings

To display the SNMP settings configured on the switch, use the following command:

```
show management
```

This command displays the following information:

- Enable / disable state for Telnet, and SNMP access
- Login statistics
  - Enable / disable state for idle timeouts
  - Maximum number of CLI sessions
- SNMP community strings

SNMPv3

SNMPv3 is an enhanced standard for SNMP that improves the security and privacy of SNMP access to managed devices and provides sophisticated control of access to the device MIB. The prior standard versions of SNMP, SNMPv1 and SNMPv2c provided no privacy and little (or no) security.

The following six RFCs provide the foundation for Extreme Networks implementation of SNMPv3:

- RFC 3410, *Introduction to version 3 of the Internet-standard Network Management Framework*, provides an overview of SNMPv3.
- RFC 3411, *An Architecture for Describing SNMP Management Frameworks*, talks about SNMP architecture, especially the architecture for security and administration.
- RFC 3412, *Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)*, talks about the message processing models and dispatching that can be a part of an SNMP engine.
- RFC 3413, *SNMPv3 Applications*, talks about the different types of applications that can be associated with an SNMPv3 engine.
- RFC 3414, *The User-Based Security Model for Version 3 of the Simple Network Management Protocol (SNMPv3)*, describes the User-Based Security Model (USM).
- RFC 3415, *View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*, talks about VACM as a way to access the MIB.

SNMPv3 Overview

The SNMPv3 standards for network management were primarily driven by the need for greater security and access control. The new standards use a modular design and model management information by cleanly defining a message processing subsystem, a security subsystem, and an access control subsystem.

The message processing (MP) subsystem helps identify the MP model to be used when processing a received Protocol Data Unit (PDU), the packets used by SNMP for communication. This layer helps in implementing a multi-lingual agent, so that various versions of SNMP can coexist simultaneously in the same network.

The security subsystem features the use of various authentication and privacy protocols with various timeliness checking and engine clock synchronization schemes. SNMPv3 is designed to be secure against:

- Modification of information, where an in-transit message is altered.
- Masquerades, where an unauthorized entity assumes the identity of an authorized entity.
- Message stream modification, where packets are delayed and / or replayed.
- Disclosure, where packet exchanges are sniffed (examined) and information is learned about the contents.

The access control subsystem provides the ability to configure whether access to a managed object in a local MIB is allowed for a remote principal. The access control scheme allows you to define access policies based on MIB views, groups, and multiple security levels.

In addition, the SNMPv3 target and notification MIBs provide a more procedural approach for the generation and filtering of notifications.

SNMPv3 objects are stored in non-volatile memory unless specifically assigned to volatile storage. Objects defined as permanent cannot be deleted or modified.

NOTE

In SNMPv3, many objects can be identified by a human-readable string or by a string of hex octets. In many commands, you can use either a character string, or a colon separated string of hex octets to specify objects. This is indicated by the keyword `hex` *used in the command.*

Message Processing

A particular network manager may require messages that conform to a particular version of SNMP. The choice of the SNMPv1, SNMPv2, or SNMPv3 message processing model can be configured for each network manager as its target address is configured. The selection of the message processing model is configured with the `mp-model` keyword in the following command:

```
configure snmpv3 add target-params {hex} <param_name> user {hex} <user_name> mp-model
[snmpv1 | snmpv2c | snmpv3] sec-model [snmpv1 | snmpv2c | usm] {sec-level [noauth |
authnopriv | priv]} {volatile}
```

SNMPv3 Security

In SNMPv3 the User-Based Security Model (USM) for SNMP was introduced. USM deals with security related aspects like authentication, encryption of SNMP messages and defining users and their various access security levels. This standard also encompass protection against message delay and message replay.

USM Timeliness Mechanisms

An Extreme switch has one SNMPv3 engine, identified by its *snmpEngineID*. The first four octets are fixed to 80:00:07:7C, which represents the Extreme Networks Vendor ID. By default, the additional octets for the snmpEngineID are generated from the device MAC address. Every SNMPv3 engine necessarily maintains two objects: *SNMPEngineBoots*, which is the number of reboots the agent has experienced and *SNMPEngineTime*, which is the engine local time since reboot. It has a local copy of these objects and the *latestReceivedEngineTime* for every authoritative engine it wants to communicate with. Comparing these objects with the values received in messages and then applying certain rules to decide upon the message validity accomplish protection against message delay or message replay.

In a chassis, the `snmpEngineID` will be generated using the MAC address of the MSM with which the switch boots first.

The *snmpEngineID* can be configured from the command line, but once the `snmpEngineID` is changed, default users will be reverted back to their original passwords/keys, while non-default users will be reset to the security level of no authorization, no privacy. To set the *snmpEngineID*, use the following command:

```
configure snmpv3 engine-id <hex_engine_id>
```

SNMPEngineBoots can also be configured from the command line. *SNMPEngineBoots* can be set to any desired value but will latch on its maximum, 2147483647. To set the *SNMPEngineBoots,* use the following command:

```
configure snmpv3 engine-boots <(1-2147483647)>
```

Users, Groups, and Security

SNMPv3 controls access and security using the concepts of users, groups, security models, and security levels.

Users. Users are created by specifying a user name. Depending on whether the user will be using authentication and/or privacy, you would also specify an authentication protocol (MD5 or SHA) with password or key, and/or privacy (DES) password or key. To create a user, use the following command:

```
configure snmpv3 add user {hex} <user_name> {authentication [md5 | sha] [hex
<hex_auth_password> | <auth_password>]} {privacy [hex <hex_priv_password> |
<priv_password>]} {volatile}
```

A number of default, permanent users are initially available. The default user names are: *admin, initial, initialmd5, initialsha, initialmd5Priv, initialshaPriv.* The default password for *admin* is *password*. For the other default users, the default password is the user name.

To display information about a user, or all users, use the following command:

```
show snmpv3 user {{hex} <user_name>}
```

To delete a user, use the following command:

```
configure snmpv3 delete user [all-non-defaults | {hex} <user_name>]
```

NOTE

The SNMPv3 specifications describe the concept of a security name. In the ExtremeWare XOS implementation, the user name and security name are identical. In this manual, both terms are used to refer to the same thing.

Groups. Groups are used to manage access for the MIB. You use groups to define the security model, the security level, and the portion of the MIB that members of the group can read or write. To underscore the access function of groups, groups are defined using the following command:

```
configure snmpv3 add access {hex} <group_name> {sec-model [snmpv1 | snmpv2 | usm]}
{sec-level [noauth | authnopriv | authpriv]} {read-view {hex} <view_name>} {write-view
{hex} <view_name>} {notify-view {hex} <view_name>} {volatile}
```

The security model and security level are discussed in “Security Models and Levels” on page 44. The view names associated with a group define a subset of the MIB (subtree) that can be accessed by members of the group. The read view defines the subtree that can be read, write view defines the subtree that can be written to, and notify view defines the subtree that notifications can originate from. MIB views are discussed in “MIB Access Control” on page 44.

A number of default (permanent) groups are already defined. These groups are: *admin, initial, v1v2c_ro, v1v2c_rw.* To display information about the access configuration of a group or all groups, use the following command:

```
show snmpv3 access {{hex} <group_name>}
```

Users are associated with groups using the following command:

```
configure snmpv3 add group {hex} <group_name> user {hex} <user_name> {sec-model
[snmpv1| snmpv2 | usm]} {volatile}
```

To show which users are associated with a group, use the following command:

```
show snmpv3 group {{hex} <group_name> {user {hex} <user_name>}}
```

To delete a group, use the following command:

```
configure snmpv3 delete access [all-non-defaults | {{hex} <group_name> {sec-model
[snmpv1 | snmpv2c | usm] sec-level [noauth | authnopriv | priv]}}]
```

When you delete a group, you do not remove the association between the group. To delete the association between a user and a group, use the following command:

```
configure snmpv3 delete group {{hex} <group_name>} user [all-non-defaults | {{hex}
<user_name> {sec-model [snmpv1|snmpv2c|usm]}}]
```

Security Models and Levels. For compatibility, SNMPv3 supports three security models:

- SNMPv1—no security
- SNMPv2c—community strings based security
- SNMPv3—USM security

The default is User-Based Security Model (USM). You can select the security model based on the network manager in your network.

The three security levels supported by USM are:

- noAuthnoPriv—No authentication, no privacy. This is the case with existing SNMPv1/v2c agents.
- AuthnoPriv—Authentication, no privacy. Messages are tested only for authentication.
- AuthPriv—Authentication, privacy. This represents the highest level of security and requires every message exchange to pass the authentication and encryption tests.

When a user is created, an authentication method is selected, and the authentication and privacy passwords or keys are entered.

When MD5 authentication is specified, HMAC-MD5-96 is used to achieve authentication with a 16-octet key, which generates an 128-bit authorization code. This code is inserted in msgAuthenticationParameters field of SNMPv3 PDUs when the security level is specified as either AuthnoPriv or AuthPriv. Specifying SHA authentication uses the HMAC-SHA protocol with a 20-octet key for authentication.

For privacy, a 16-octet key is provided as input to DES-CBS encryption protocol, which generates an encrypted PDU to be transmitted. DES uses bytes 1-7 to make a 56 bit key. This key (encrypted itself) is placed in msgPrivacyParameters of SNMPv3 PDUs when the security level is specified as AuthPriv.

MIB Access Control

SNMPv3 provides a fine-grained mechanism for defining which parts of the MIB can be accessed. This is referred to as the View-Based Access Control Model (VACM).

MIB views represent the basic building blocks of VACM. They are used to define a subset of the information in the MIB. Access to read, to write, and to generate notifications is based on the

relationship between a MIB view and an access group. The users of the access group can then read, write, or receive notifications from the part of the MIB defined in the MIB view as configured in the access group.

A view name, a MIB subtree / mask, and an inclusion or exclusion define every MIB view. For example, there is a *System* group defined under the MIB-2 tree. The Object Identifier (OID) for MIB-2 is 1.3.6.1.2, and the *System* group is defined as MIB-2.1.1, or directly as 1.3.6.1.2.1.1.

To define a MIB view which includes only the *System* group, use the following subtree / mask combination:

```
1.3.6.1.2.1.1 / 1.1.1.1.1.1.1.0
```

The mask can also be expressed in hex notation (this is used for the ExtremeWare XOS CLI):

```
1.3.6.1.2.1.1 / fe
```

To define a view that includes the entire MIB-2, use the following subtree / mask:

```
1.3.6.1.2.1.1 / 1.1.1.1.1.0.0.0
```

which, on the command line, is:

```
1.3.6.1.2.1.1 / f8
```

When you create the MIB view, you can choose to include the MIB subtree / mask, or to exclude the MIB subtree / mask. To create a MIB view, use the following command:

```
configure snmpv3 add mib-view {hex} <view_name> subtree <object_identifier>
{/<subtree_mask>} {type [included | excluded]} {volatile}
```

After the view has been created, you can repeatedly use the `configure snmpv3 add mib-view` command to include and / or exclude MIB subtree / mask combinations to precisely define the items you wish to control access to.

In addition to the user-created MIB views, there are three default views. They are of storage type permanent and cannot be deleted, but they can be modified. The default views are: *defaultUserView*, *defaultAdminView*, and *defaultNotifyView*. To show MIB views, use the following command:

```
show snmpv3 mib-view {{hex} <view_name> {subtree <object_identifier>}}
```

To delete a MIB view, use the following command:

```
configure snmpv3 delete mib-view [all-non-defaults | {{hex} <view_name> {subtree
<object_identifier>}}]
```

MIB views which are being used by security groups cannot be deleted.

Notification

SNMPv3 notification is an enhancement to the concept of SNMP traps. Notifications are messages sent from an agent to the network manager, typically in response to some state change on the agent system. With SNMPv3, you can define precisely which traps you want sent, to which receiver by defining filter profiles to use for the notification receivers.

To configure notifications, you will configure a target address for the process that receives the notification, a target parameters name, and a list of notification tags. The target parameters specify the security and message processing models to use for the notifications to the target. The target parameters

name also points to the filter profile used to filter the notifications. Finally, the notification tags are added to a notification table so that any target addresses using that tag will receive notifications.

Target Addresses

A target address is similar to the earlier concept of a trap receiver. To configure a target address, use the following command:

```
configure snmpv3 add target-addr {hex} <addr_name> param {hex} <param_name> ipaddress
<ip_address> {volatile}
```

In configuring the target address you supply an address name that identifies the target address, a parameters name that indicates the message processing model and security for the messages sent to the target address, and the IP address and port for the receiver. The parameters name also is used to indicate the filter profile used for notifications. The target parameters is discussed in "Target Parameters" next.

To display target addresses, use the following command:

```
show snmpv3 target-addr {{hex} <addr_name>}
```

To delete a single target address or all target addresses, use the following command:

```
configure snmpv3 delete target-addr [{{hex} <addr_name>} | all]
```

Target Parameters

Target parameters specify the message processing model, security model, security level, and user name (security name) used for messages sent to the target address. See "Message Processing" on page 42 and "Users, Groups, and Security" on page 43 for more details on these topics. In addition, the target parameter name used for a target address points to a filter profile used to filter notifications. When you specify a filter profile, you associate it with a parameter name, so you need to create different target parameter names if you use different filters for different target addresses.

To create a target parameter name, and set the message processing and security settings associated with it, use the following command:

```
configure snmpv3 add target-params {hex} <param_name> user {hex} <user_name> mp-model
[snmpv1 | snmpv2c | snmpv3] sec-model [snmpv1 | snmpv2c | usm] {sec-level [noauth |
authnopriv | priv]} {volatile}
```

To display the options associated with a target parameters name, or all target parameters names, use the following command:

```
show snmpv3 target-params {{hex} <target_params>}
```

To delete one or all the target parameters, use the following command:

```
configure snmpv3 delete target-params [{{hex} <param_name>} | all]
```

Filter Profiles and Filters

A filter profile is a collection of filters that specifies which notifications should be sent to a target address. A filter is defined by a MIB subtree and mask, and by whether that subtree and mask is included or excluded from notification.

When you create a filter profile, you are only associating a filter profile name with a target parameter name. The filters that make up the profile are created and associated with the profile using a different command. To create a filter profile, use the following command:

```
configure snmpv3 add filter-profile {hex} <profile_name> param {hex} <param_name>
{volatile}
```

After the profile name has been created, you can associate filters with it using the following command:

```
configure snmpv3 add filter {hex} <profile_name> subtree <object_identifier>
{/<subtree_mask>} type [included | excluded] {volatile}
```

The MIB subtree and mask are discussed in “MIB Access Control” on page 44, as filters are closely related to MIB views. You can add filters together, including and excluding different subtrees of the MIB until your filter meets your needs.

To display the association between parameter names and filter profiles, use the following command:

```
show snmpv3 filter-profile {{hex} <profile_name>} {param {hex} <param_name>}
```

To display the filters that belong a filter profile, use the following command:

```
show snmpv3 filter {{hex} <profile_name> {{subtree} <object_identifier>}
```

To delete a filter or all filters from a filter profile, use the following command:

```
configure snmpv3 delete filter [all | [{hex} <profile_name> {subtree
<object_identifier>}]]
```

To remove the association of a filter profile or all filter profiles with a parameter name, use the following command:

```
configure snmpv3 delete filter-profile [all |[{hex} <profile_name> {param {hex}
<param_name>}]]
```

Notification Tags

When you create a target address, you associate a list of notification tags with the target, or by default, the *defaultNotify* tag is associated with the target. When notifications are generated, only targets associated with tags currently in an internal structure, called *snmpNotifyTable*, will be notified. To add an entry to the table, use the following command:

```
configure snmpv3 add notify {hex} <notify_name> tag {hex} <tag> {volatile}
```

Any targets associated with tags in the *snmpNotifyTable* will be notified, based on the filter profile associated with the target.

To display the notifications that are set, use the following command:

```
show snmpv3 notify {{hex} <notify_name>}
```

To delete an entry from the *snmpNotifyTable*, use the following command:

```
configure snmpv3 delete notify [{{hex} <notify_name>} | all-non-defaults]
```

You cannot delete the default entry from the table, so any targets configured with the *defaultNotify* tag will always receive notifications consistent with any filter profile specified.

Configuring Notifications

Since the target parameters name is used to point to a number of objects used for notifications, configure the target parameter name entry first. You can then configure the target address, filter profiles and filters, and any necessary notification tags.

Authenticating Users

ExtremeWare XOS provides two methods to authenticate users who login to the switch:

- RADIUS client
- TACACS+

NOTE

You cannot configure RADIUS and TACACS+ at the same time.

RADIUS Client

Remote Authentication Dial In User Service (RADIUS, RFC 2138) is a mechanism for authenticating and centrally administrating access to network nodes. The ExtremeWare XOS RADIUS client implementation allows authentication for Telnet or console access to the switch.

TACACS+

Terminal Access Controller Access Control System Plus (TACACS+) is a mechanism for providing authentication, authorization, and accounting on a centralized server, similar in function to the RADIUS client. The ExtremeWare XOS version of TACACS+ is used to authenticate prospective users who are attempting to administer the switch. TACACS+ is used to communicate between the switch and an authentication database.

Configuring RADIUS Client and TACACS+

For detailed information about configuring a RADIUS client or TACACS+, see Chapter 9.

Using the Simple Network Time Protocol

ExtremeWare XOS supports the client portion of the Simple Network Time Protocol (SNTP) Version 3 based on RFC1769. SNTP can be used by the switch to update and synchronize its internal clock from a Network Time Protocol (NTP) server. When enabled, the switch sends out a periodic query to the indicated NTP server, or the switch listens to broadcast NTP updates. In addition, the switch supports the configured setting for Greenwich Mean time (GMT) offset and the use of Daylight Saving Time. These features have been tested for year 2000 compliance.

Configuring and Using SNTP

To use SNTP, follow these steps:

1. Identify the host(s) that are configured as NTP server(s). Additionally, identify the preferred method for obtaining NTP updates. The options are for the NTP server to send out broadcasts, or for switches using NTP to query the NTP server(s) directly. A combination of both methods is possible. You must identify the method that should be used for the switch being configured.
2. Configure the Greenwich Mean Time (GMT) offset and Daylight Saving Time preference. The command syntax to configure GMT offset and usage of Daylight Saving Time is as follows:

   ```
   configure timezone {name <std_timezone_ID>} <GMT_offset>
   {autodst {name <dst_timezone_ID>} {<dst_offset>}
   {begins [every <floatingday> | on <absoluteday>] {at <time_of_day_hour>
   <time_of_day_minutes>}
   {ends [every <floatingday> | on <absoluteday>] {at <time_of_day_hour>
   <time_of_day_minutes>}}}
   ```

   By default, Daylight Saving Time is assumed to begin on the first Sunday in April at 2:00 AM, and end the last Sunday in October at 2:00 AM, and be offset from standard time by one hour. If this is the case in your timezone, you can set up automatic daylight savings adjustment with the command:

   ```
   configure timezone <GMT_offset> autodst
   ```

   If your timezone uses starting and ending dates and times that differ from the default, you can specify the starting and ending date and time in terms of a floating day, as follows:

   ```
   configure timezone name MET 60 autodst name MDT begins every last sunday march at
   1 30 ends every last sunday october at 1 30
   ```

   You can also specify a specific date and time, as shown in the following command.

   ```
   configure timezone name NZST 720 autodst name NZDT 60 begins every first sunday
   october at 2 00 ends on 3 16 2004 at 2 00
   ```

   The optional timezone IDs are used to identify the timezone in display commands such as `show switch {detail}`.

   Table 8 describes the command options in detail.

Table 8: Time zone configuration command options

GMT_offset	Specifies a Greenwich Mean Time (GMT) offset, in + or - minutes.
std-timezone-ID	Specifies an optional name for this timezone specification. May be up to six characters in length. The default is an empty string.
autodst	Enables automatic Daylight Savings Time.
dst-timezone-ID	Specifies an optional name for this DST specification. May be up to six characters in length. The default is an empty string.
dst_offset	Specifies an offset from standard time, in minutes. Value is in the range of 1 to 60. Default is 60 minutes.
floating_day	Specifies the day, week, and month of the year to begin or end DST each year. Format is: <week> <day> <month> where: • <week> is specified as [first | second | third | fourth | last] • <day> is specified as [sunday | monday | tuesday | wednesday | thursday | friday | saturday] • <month> is specified as [january | february | march | april | may | june | july | august | september | october | november | december] Default for beginning is first sunday april; default for ending is last sunday october.

Table 8: Time zone configuration command options (continued)

absolute_day	Specifies a specific day of a specific year on which to begin or end DST. Format is: <month> <day> <year> where: • <month> is specified as 1-12 • <day> is specified as 1-31 • <year> is specified as 1970 - 2035 The year must be the same for the begin and end dates.
time_of_day_hour	Specifies the time of day to begin or end Daylight Savings Time. May be specified as an hour (0-23). Default is 2.
time_of_day_minutes	Specify the minute to begin or end Daylight Savings Time. May be specified as a minute (0-59).
noautodst	Disables automatic Daylight Savings Time.

Automatic Daylight Savings Time (DST) changes can be enabled or disabled. The default setting is enabled. To disable automatic DST, use the command:

```
configure timezone {name <std_timezone_ID>} <GMT_offset> noautodst
```

3 Enable the SNTP client using the following command:

```
enable sntp-client
```

After it has been enabled, the switch sends out a periodic query to the NTP servers defined later (if configured) or listens to broadcast NTP updates from the network. The network time information is automatically saved into the on-board real-time clock.

4 If you would like this switch to use a directed query to the NTP server, configure the switch to use the NTP server(s). If the switch listens to NTP broadcasts, skip this step. To configure the switch to use a directed query, use the following command:

```
configure sntp-client [primary | secondary] <host-name-or-ip>
```

NTP queries are first sent to the primary server. If the primary server does not respond within 1 second, or if it is not synchronized, the switch queries the secondary server (if one is configured). If the switch cannot obtain the time, it restarts the query process. Otherwise, the switch waits for the `sntp-client update interval` before querying again.

5 Optionally, the interval for which the SNTP client updates the real-time clock of the switch can be changed using the following command:

```
configure sntp-client update-interval <update-interval>
```

The default `sntp-client update-interval` value is 64 seconds.

6 You can verify the configuration using the following commands:

— `show sntp-client`

This command provides configuration and statistics associated with SNTP and its connectivity to the NTP server.

— `show switch {detail}`

This command indicates the GMT offset, the Daylight Savings Time configuration and status, and the current local time.

NTP updates are distributed using GMT time. To properly display the local time in logs and other timestamp information, the switch should be configured with the appropriate offset to GMT based on geographical location. Table 9 describes GMT offsets.

Table 9: Greenwich mean time offsets

GMT Offset in Hours	GMT Offset in Minutes	Common Time Zone References	Cities
+0:00	+0	GMT - Greenwich Mean	London, England; Dublin, Ireland; Edinburgh, Scotland; Lisbon, Portugal; Reykjavik, Iceland; Casablanca, Morocco
		UT or UTC - Universal (Coordinated)	
		WET - Western European	
-1:00	-60	WAT - West Africa	Azores, Cape Verde Islands
-2:00	-120	AT - Azores	
-3:00	-180		Brasilia, Brazil; Buenos Aires, Argentina; Georgetown, Guyana;
-4:00	-240	AST - Atlantic Standard	Caracas; La Paz
-5:00	-300	EST - Eastern Standard	Bogota, Columbia; Lima, Peru; New York, NY, Trevor City, MI USA
-6:00	-360	CST - Central Standard	Mexico City, Mexico
-7:00	-420	MST - Mountain Standard	Saskatchewan, Canada
-8:00	-480	PST - Pacific Standard	Los Angeles, CA, Cupertino, CA, Seattle, WA USA
-9:00	-540	YST - Yukon Standard	
-10:00	-600	AHST - Alaska-Hawaii Standard	
		CAT - Central Alaska	
		HST - Hawaii Standard	
-11:00	-660	NT - Nome	
-12:00	-720	IDLW - International Date Line West	
+1:00	+60	CET - Central European	Paris, France; Berlin, Germany; Amsterdam, The Netherlands; Brussels, Belgium; Vienna, Austria; Madrid, Spain; Rome, Italy; Bern, Switzerland; Stockholm, Sweden; Oslo, Norway
		FWT - French Winter	
		MET - Middle European	
		MEWT - Middle European Winter	
		SWT - Swedish Winter	
+2:00	+120	EET - Eastern European, Russia Zone 1	Athens, Greece; Helsinki, Finland; Istanbul, Turkey; Jerusalem, Israel; Harare, Zimbabwe
+3:00	+180	BT - Baghdad, Russia Zone 2	Kuwait; Nairobi, Kenya; Riyadh, Saudi Arabia; Moscow, Russia; Tehran, Iran
+4:00	+240	ZP4 - Russia Zone 3	Abu Dhabi, UAE; Muscat; Tblisi; Volgograd; Kabul
+5:00	+300	ZP5 - Russia Zone 4	
+5:30	+330	IST – India Standard Time	New Delhi, Pune, Allahabad, India
+6:00	+360	ZP6 - Russia Zone 5	
+7:00	+420	WAST - West Australian Standard	
+8:00	+480	CCT - China Coast, Russia Zone 7	
+9:00	+540	JST - Japan Standard, Russia Zone 8	
+10:00	+600	EAST - East Australian Standard	
		GST - Guam Standard	
		Russia Zone 9	

Table 9: Greenwich mean time offsets (continued)

GMT Offset in Hours	GMT Offset in Minutes	Common Time Zone References	Cities
+11:00	+660		
+12:00	+720	IDLE - International Date Line East NZST - New Zealand Standard NZT - New Zealand	Wellington, New Zealand; Fiji, Marshall Islands

SNTP Example

In this example, the switch queries a specific NTP server and a backup NTP server. The switch is located in Cupertino, CA, and an update occurs every 20 minutes. The commands to configure the switch are as follows:

```
configure timezone -480 autodst
configure sntp-client update-interval 1200
enable sntp-client
configure sntp-client primary 10.0.1.1
configure sntp-client secondary 10.0.1.2
```

4 Configuring Slots and Ports on a Switch

This chapter covers the following topics:

- Configuring a Slot on a Modular Switch on page 53
- Configuring Ports on a Switch on page 54
- Jumbo Frames on page 56
- Load Sharing on the Switch on page 58
- Switch Port-Mirroring on page 59
- Extreme Discovery Protocol on page 60

Configuring a Slot on a Modular Switch

If a slot has not been configured for a particular type of module, then any type of module is accepted in that slot, and a default port and VLAN configuration is automatically generated.

After any port on the module has been configured (for example, a VLAN association, a VLAN tag configuration, or port parameters), all the port information and the module type for that slot must be saved to non-volatile storage. Otherwise, if the modular switch is rebooted or the module is removed from the slot, the port, VLAN, and module configuration information is not saved.

NOTE

For information on saving the configuration, see Appendix A.

You can configure the modular switch with the type of I/O module that is installed in each slot. To do this, use the following command:

```
configure slot <slot> module <module_type>
```

You can also preconfigure the slot before inserting the module. This allows you to begin configuring the module and ports before installing the module in the chassis.

If a slot is configured for one type of module, and a different type of module is inserted, the inserted module is put into a mismatch state, and is not brought online. To use the new module type in a slot, the slot configuration must be cleared or configured for the new module type. To clear the slot of a previously assigned module type, use the following command:

```
clear slot <slot>
```

All configuration information related to the slot and the ports on the module is erased. If a module is present when you issue this command, the module is reset to default settings.

To display information about a particular slot, use the following command:

```
show slot <slot number>
```

Information displayed includes:

- Card type, serial number, part number.
- Current state (power down, operational, diagnostic, mismatch).
- Port information.

If no slot is specified, information for all slots is displayed.

Configuring Ports on a Switch

On a modular switch, the port number is a combination of the slot number and the port number. The nomenclature for the port number is as follows:

```
slot:port
```

For example, if an I/O module that has a total of four ports is installed in slot 2 of the chassis, the following ports are valid:

- `2:1`
- `2:2`
- `2:3`
- `2:4`

You can also use wildcard combinations (*) to specify multiple modular slot and port combinations. The following wildcard combinations are allowed:

- `slot:*`—Specifies all ports on a particular I/O module.
- `slot:`*`x`*`-slot:`*`y`*—Specifies a contiguous series of ports on a particular I/O module.
- `slot`*`a`*`:`*`x`*`-slot`*`b`*`:`*`y`*—Specifies a contiguous series of ports that begin on one I/O module and end on another I/O module.

Enabling and Disabling Switch Ports

By default, all ports are enabled. To enable or disable one or more ports on a modular switch, use the following command:

```
enable port [<port_list> | all]
disable port [<port_list> | all]
```

For example, to disable slot 7, ports 3, 5, and 12 through 15 on a modular switch, use the following command:

```
disable ports 7:3,7:5,7:12-7:15
```

Even though a port is disabled, the link remains enabled for diagnostic purposes.

Configuring Switch Port Speed and Duplex Setting

By default, the switch is configured to use autonegotiation to determine the port speed and duplex setting for each port. You can manually configure the duplex setting and the speed of 10/100 Mbps ports, and you can manually configure the duplex setting of Gigabit Ethernet ports.

10BASE-T and 100BASE-TX ports can connect to either 10BASE-T or 100BASE-T networks. By default, the ports autonegotiate port speed. You can also configure each port for a particular speed (either 10 Mbps or 100 Mbps).

Gigabit Ethernet ports are statically set to 1 Gbps, and their speed cannot be modified.

To configure port speed and duplex setting, use the following command:

```
configure ports <port_list> auto off {speed [10 | 100 | 1000]} duplex [half | full]
```

To configure the system to autonegotiate, use the following command:

```
configure ports <port_list> auto on
```

Flow control is fully supported only on Gigabit Ethernet ports. Gigabit ports both advertise support and respond to pause frames. 10/100 Mbps Ethernet ports also respond to pause frames, but do not advertise support. Neither 10/100 Mbps or Gigabit Ethernet ports initiate pause frames.

Flow Control is enabled or disabled as part of autonegotiation. If autonegotiation is set to off, flow control is disabled. When autonegotiation is turned on, flow control is enabled. ExtremeWare XOS does not support turning off autonegotiation on the management port.

Turning Off Autonegotiation for a Gigabit Ethernet Port

In certain interoperability situations, you may need to turn autonegotiation off on a Gigabit Ethernet port. Even though a Gigabit Ethernet port runs only at full duplex, you must specify the duplex setting.

NOTE

1000BASE-TX ports support only autonegotiation.

The following example turns autonegotiation off for port 1 (a Gigabit Ethernet port) on a module located in slot 1 of a modular switch:

```
configure ports 1:1 auto off duplex full
```

The following example turns autonegotiation off for port 4 (a Gigabit Ethernet port) on a stand-alone switch:

```
configure ports 4 auto off duplex full
```

Table 10 lists the support for autonegotiation, speed, and duplex setting for the various types of ports.

Table 10: Support for Autonegotiation on Various Ports

PHY	Autonegotiation	Speed	Duplex
10 G	Not configurable; On	10 G	Not configurable

Table 10: Support for Autonegotiation on Various Ports

PHY	Autonegotiation	Speed	Duplex
1 G fiber	On/Off	1 G	Not configurable; Full duplex
1 G copper at 1000 Mbps	Not configurable	1 G	Not configurable
1 G copper at 10/100 Mbps	On/Off	10/100 Mbps	Full/Half duplex

Jumbo Frames

Jumbo frames are Ethernet frames that are larger than 1522 bytes, including four bytes used for the cyclic redundancy check (CRC). Extreme products support switching and routing of jumbo frames at wire-speed on all ports.

Jumbo frames are used between endstations that support larger frame sizes for more efficient transfers of bulk data. Both endstations involved in the transfer must be capable of supporting jumbo frames. The switch only performs IP fragmentation, or participates in maximum transmission unit (MTU) negotiation on behalf of devices that support jumbo frames.

Enabling Jumbo Frames

To enable jumbo frame support, enable jumbo frames on the desired ports. To set the maximum jumbo frame size, use the following command:

```
configure jumbo-frame size <number>
```

The jumbo frame size range is 1523 to 9216. This value describes the maximum size of the frame in transit (on the wire), and includes 4 bytes of CRC plus another 4 bytes if 802.1Q tagging is being used.

Next, enable support on the physical ports that will carry jumbo frames using the following command:

```
enable jumbo-frame ports [<port_list> | all]
```

NOTE

Some network interface cards (NICs) have a configured maximum MTU size that does not include the additional 4 bytes of CRC. Ensure that the NIC maximum MTU size is at or below the maximum MTU size configured on the switch. Frames that are larger than the MTU size configured on the switch are dropped at the ingress port.

Path MTU Discovery

Using path MTU discovery, a source host assumes that the path MTU is the MTU of the first hop (which is known). The host sends all datagrams on that path with the “don’t fragment” (DF) bit set, which restricts fragmentation. If any of the datagrams must be fragmented by an Extreme switch along the path, the Extreme switch discards the datagrams and returns an ICMP Destination Unreachable message to the sending host, with a code meaning “fragmentation needed and DF set”. When the source host receives the message (sometimes called a “Datagram Too Big” message), the source host reduces its assumed path MTU and retransmits the datagrams.

The path MTU discovery process ends when one of the following is true:

- The source host sets the path MTU low enough that its datagrams can be delivered without fragmentation.
- The source host does not set the DF bit in the datagram headers.

If it is willing to have datagrams fragmented, a source host can choose not to set the DF bit in datagram headers. Normally, the host continues to set DF in all datagrams, so that if the route changes and the new path MTU is lower, the host can perform path MTU discovery again.

IP Fragmentation with Jumbo Frames

ExtremeWare XOS supports the fragmenting of IP packets. If an IP packet originates in a local network that allows large packets and those packets traverse a network that limits packets to a smaller size, the packets are fragmented instead of discarded.

This feature is designed to be used in conjunction with jumbo frames. Frames that are fragmented are not processed at wire-speed within the switch fabric.

NOTE

Jumbo frame-to-jumbo frame fragmentation is not supported. Only jumbo frame-to-normal frame fragmentation is supported.

To configure VLANs for IP fragmentation:

1. Enable jumbo frames on the incoming port.
2. Add the port to a VLAN.
3. Assign an IP address to the VLAN.
4. Enable ipforwarding on the VLAN.

IP Fragmentation within a VLAN

ExtremeWare XOS supports IP fragmentation within a VLAN. This feature does not require you to configure the MTU size. To use IP fragmentation within a VLAN:

1. Enable jumbo frames on the incoming port.
2. Add the port to a VLAN.
3. Assign an IP address to the VLAN.
4. Enable ipforwarding on the VLAN.

If you leave the MTU size configured to the default value, when you enable jumbo frame support on a port on the VLAN you will receive a warning that the ip-mtu size for the VLAN is not set at maximum jumbo frame size. You can ignore this warning if you want IP fragmentation within the VLAN, only. However, if you do not use jumbo frames, IP fragmentation can only be used for traffic that stays within the same VLAN. For traffic that is set to other VLANs, to use IP fragmentation, all ports in the VLAN must be configured for jumbo frame support.

Load Sharing on the Switch

Load sharing allows you to increase bandwidth and availability by using a group of ports to carry traffic in parallel between switches. Load sharing allows the switch to use multiple ports as a single logical port. For example, VLANs see the load-sharing group as a single logical port. Most load-sharing algorithms guarantee packet sequencing between clients.

If a port in a load-sharing group fails, traffic is redistributed to the remaining ports in the load-sharing group. If the failed port becomes active again, traffic is redistributed to include that port.

NOTE

Load sharing must be enabled on both ends of the link or a network loop may result. The load-sharing types (dynamic, static) must match, but the load-sharing algorithms do not need to be the same on both ends.

Configuring Switch Load Sharing

To set up a switch to load share among ports, you must create a load-sharing group of ports. The first port in the load-sharing group is configured to be the “master” logical port. This is the reference port used in configuration commands. It can be thought of as the logical port representing the entire port group.

All the ports in a load-sharing group must have the same exact configuration, including auto negotiation, duplex setting, and so on. All the ports in a load-sharing group must also be of the same bandwidth class.

The following rules apply:

- One group can contain up to 16 ports.
- The ports in the group do not need to be contiguous.
- A load share group that spans multiple modules must use ports that are all of the same maximum bandwidth capability.

To define a load-sharing group, you assign a group of ports to a single, logical port number. To enable or disable a load-sharing group, use the following commands:

```
enable sharing <master_port> grouping <port_list> {algorithm port-based}

disable sharing <master_port>
```

NOTE

Do not disable a port that is part of a load-sharing group. Disabling the port prevents it from forwarding traffic, but still allows the link to initialize. As a result, a partner switch does not receive a valid indication that the port is not in a forwarding state, and the partner switch will continue to forward packets.

Adding and Deleting Ports in a Load-Sharing Group

Ports can be added or deleted dynamically in a load-sharing group. To add or delete ports from a load-sharing group, use the following commands:

```
configure sharing <master_port> add ports <port_list>
```

```
configure sharing <master_port> delete ports <port_list>
```

Load-Sharing Examples

This section provides examples of how to define load-sharing on modular and stand-alone switches.

Cross-Module Load Sharing on a Modular Switch

The following example defines a load-sharing group that contains ports 9 through 12 on slot 3, ports 7 through 10 on slot 5, and uses the first port in the slot 3 group as the master logical port 9:

```
enable sharing 3:9 grouping 3:9-3:12, 5:7-5:10
```

In this example, logical port 3:9 represents physical ports 3:9 through 3:12 and 5:7 through 5:10.

When using load sharing, you should always reference the master logical port of the load-sharing group (port 3:9 in the previous example) when configuring or viewing VLANs. VLANs configured to use other ports in the load-sharing group will have those ports deleted from the VLAN when load sharing becomes enabled.

Single-Module Load Sharing on a Modular Switch

Single-module load sharing is supported on all modular switches. The following example defines a load-sharing group that contains ports 9 through 12 on slot 3 and uses the first port as the master logical port 9:

```
enable sharing 3:9 grouping 3:9-3:12
```

In this example, logical port 3:9 represents physical ports 3:9 through 3:12.

When using load sharing, you should always reference the master logical port of the load-sharing group (port 3:9 in the previous example) when configuring or viewing VLANs. VLANs configured to use other ports in the load-sharing group will have those ports deleted from the VLAN when load sharing becomes enabled.

Verifying the Load-Sharing Configuration

The screen output resulting from the `show ports sharing` command lists the ports that are involved in load sharing and the master logical port identity.

Switch Port-Mirroring

Port-mirroring configures the switch to copy all traffic associated with one or more ports. The monitor port can be connected to a network analyzer or RMON probe for packet analysis. The system uses a traffic filter that copies a group of traffic to the monitor port.

The traffic filter can be defined based on one of the following criteria:

- **Physical port**—All data that traverses the port, regardless of VLAN configuration, is copied to the monitor port.

Up to eight mirroring filters and one monitor port can be configured. After a port has been specified as a monitor port, it cannot be used for any other function.

NOTE

Frames that contain errors are not mirrored.

Modular Switch Port-Mirroring Example

The following example selects slot 7, port 3 as the mirror port, and sends all traffic coming into or out of a modular switch on slot 7, port 1 to the mirror port:

```
configure mirroring add port 7:1
enable mirroring to port 7:3
```

The following example sends all traffic coming into or out of the system on slot 8, port 1 to the mirror port:

```
enable mirroring to port 8:4
configure mirroring add port 8:1
```

Extreme Discovery Protocol

The Extreme Discovery Protocol (EDP) is used to gather information about neighbor Extreme Networks switches. EDP is used to by the switches to exchange topology information. Information communicated using EDP includes:

- Switch MAC address (switch ID)
- Switch software version information
- Switch IP address
- Switch VLAN-IP information
- Switch port number

EDP is enabled on all ports by default. EDP enabled ports advertise information about the Extreme switch to other switches on the interface, and receives advertisements from other Extreme switches. Information about other Extreme switches is discarded after a timeout interval is reached without receiving another advertisement.

To disable EDP on one or more ports, use the following command:

```
disable edp ports [<ports> | all]
```

To enable EDP on specified ports, use the following command:

```
enable edp ports [<ports> | all]
```

To view EDP port information on the switch, use the following command:

```
show edp {ports [all | <ports>] {detail}}
```

To configure the advertisement interval and the timeout interval, use the following command:

```
configure edp advertisment-interval <timer> holddown-interval <timeout>
```

5 Virtual LANs (VLANs)

This chapter covers the following topics:

- Overview of Virtual LANs on page 61
- Types of VLANs on page 62
- VLAN Names on page 70
- Configuring VLANs on the Switch on page 71
- Displaying VLAN Settings on page 72

Setting up Virtual Local Area Networks (VLANs) on the switch eases many time-consuming tasks of network administration while increasing efficiency in network operations.

Overview of Virtual LANs

The term “VLAN” is used to refer to a collection of devices that communicate as if they were on the same physical LAN. Any set of ports (including all ports on the switch) is considered a VLAN. LAN segments are not restricted by the hardware that physically connects them. The segments are defined by flexible user groups you create with the command line interface.

Benefits

Implementing VLANs on your networks has the following advantages:

- **VLANs help to control traffic—**With traditional networks, congestion can be caused by broadcast traffic that is directed to all network devices, regardless of whether they require it. VLANs increase the efficiency of your network because each VLAN can be set up to contain only those devices that must communicate with each other.
- **VLANs provide extra security—**Devices within each VLAN can only communicate with member devices in the same VLAN. If a device in VLAN *Marketing* must communicate with devices in VLAN *Sales*, the traffic must cross a routing device.
- **VLANs ease the change and movement of devices—**With traditional networks, network administrators spend much of their time dealing with moves and changes. If users move to a different subnetwork, the addresses of each endstation must be updated manually.

Types of VLANs

VLANs can be created according to the following criteria:

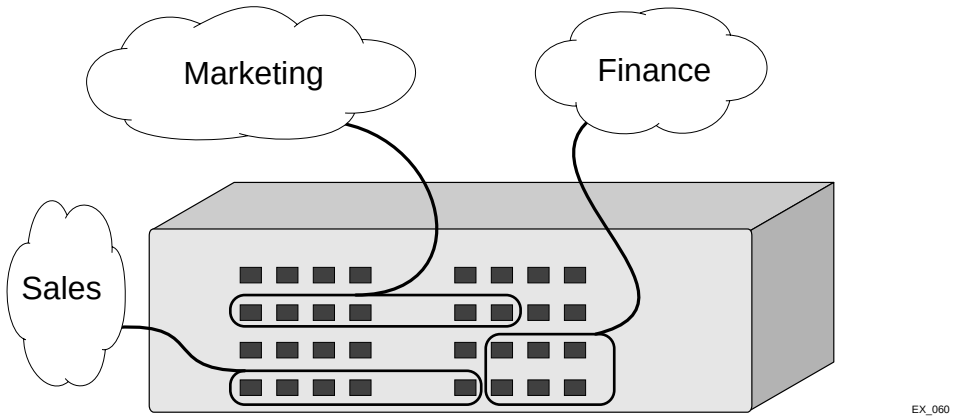
- Physical port
- 802.1Q tag
- Ethernet, LLC SAP, or LLC/SNAP Ethernet protocol type
- MAC address
- A combination of these criteria

Port-Based VLANs

In a port-based VLAN, a VLAN name is given to a group of one or more ports on the switch. All ports are members of the port-based VLAN *default*. Before you can add any port to another port-based VLAN, you must remove it from the default VLAN, unless the new VLAN uses a protocol other than the default protocol *any*. A port can be a member of only one port-based VLAN.

On the Extreme switch in Figure 1, ports 9 through 14 are part of VLAN *Marketing*; ports 25 through 29 are part of VLAN *Sales*; and ports 21 through 24 and 30 through 32 are in VLAN *Finance*.

Figure 1: Example of a port-based VLAN on an Extreme switch



For the members of the different IP VLANs to communicate, the traffic must be routed by the switch, even if they are physically part of the same I/O module. This means that each VLAN must be configured as a router interface with a unique IP address.

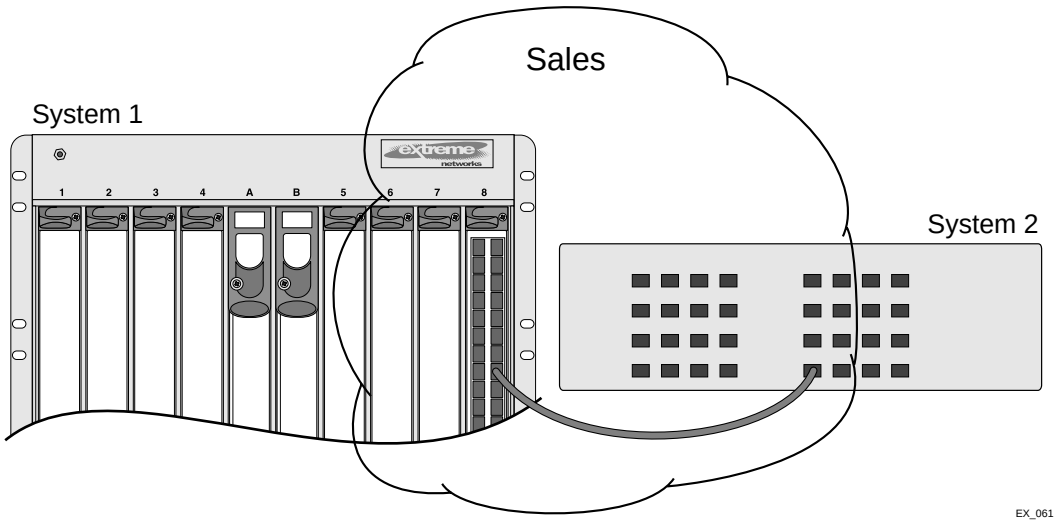
Spanning Switches with Port-Based VLANs

To create a port-based VLAN that spans two switches, you must do two things:

1. Assign the port on each switch to the VLAN.
2. Cable the two switches together using one port on each switch per VLAN.

Figure 2 illustrates a single VLAN that spans a BlackDiamond switch and another Extreme switch. All ports on the BlackDiamond switch belong to VLAN *Sales*. Ports 1 through 29 on the other Extreme switch also belong to VLAN *Sales*. The two switches are connected using slot 8, port 4 on system 1 (the BlackDiamond switch), and port 29 on system 2 (the other switch).

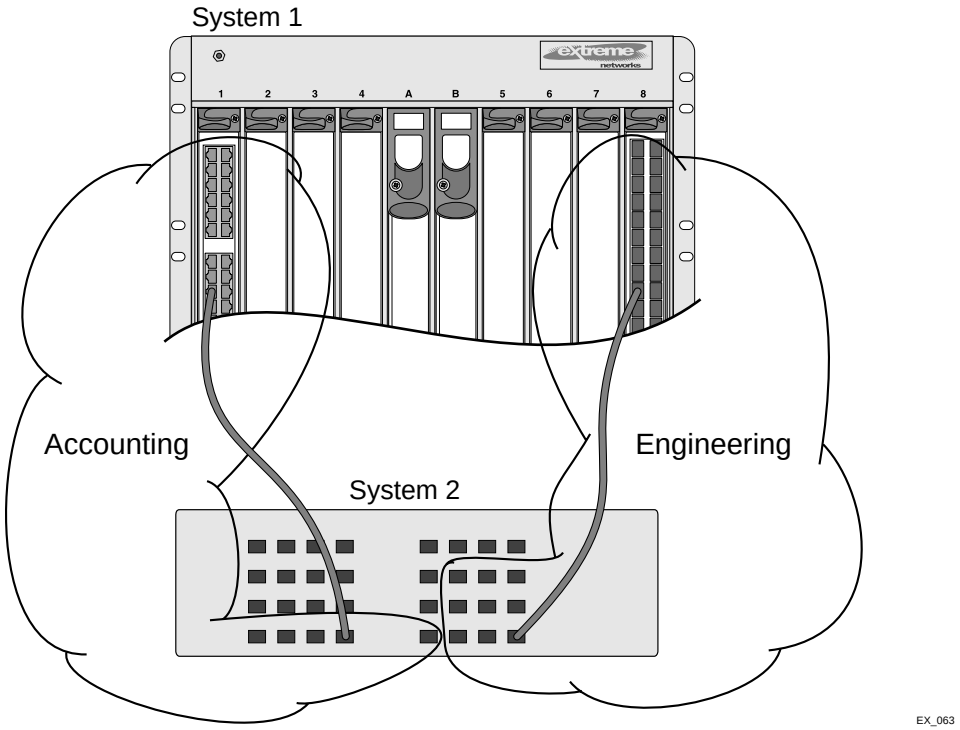
Figure 2: Single port-based VLAN spanning two switches



To create multiple VLANs that span two switches in a port-based VLAN, a port on system 1 must be cabled to a port on system 2 for each VLAN you want to have span across the switches. At least one port on each switch must be a member of the corresponding VLANs, as well.

Figure 3 illustrates two VLANs spanning two switches. On system 2, ports 25 through 29 are part of VLAN *Accounting*; ports 21 through 24 and ports 30 through 32 are part of VLAN *Engineering*. On system 1, all port on slot 1 are part of VLAN *Accounting*; all ports on slot 8 are part of VLAN *Engineering*.

Figure 3: Two port-based VLANs spanning two switches



VLAN *Accounting* spans system 1 and system 2 by way of a connection between system 2, port 29 and system 1, slot 1, port 6. VLAN *Engineering* spans system 1 and system 2 by way of a connection between system 2, port 32, and system 1, slot 8, port 6.

Using this configuration, you can create multiple VLANs that span multiple switches, in a daisy-chained fashion. Each switch must have a dedicated port for each VLAN. Each dedicated port must be connected to a port that is a member of its VLAN on the next switch.

Tagged VLANs

Tagging is a process that inserts a marker (called a *tag*) into the Ethernet frame. The tag contains the identification number of a specific VLAN, called the *VLANid*.

NOTE

The use of 802.1Q tagged packets may lead to the appearance of packets slightly bigger than the current IEEE 802.3/Ethernet maximum of 1,518 bytes. This may affect packet error counters in other devices, and may also lead to connectivity problems if non-802.1Q bridges or routers are placed in the path.

Uses of Tagged VLANs

Tagging is most commonly used to create VLANs that span switches. The switch-to-switch connections are typically called *trunks*. Using tags, multiple VLANs can span multiple switches using one or more trunks. In a port-based VLAN, each VLAN requires its own pair of trunk ports, as shown in Figure 3. Using tags, multiple VLANs can span two switches with a single trunk.

Another benefit of tagged VLANs is the ability to have a port be a member of multiple VLANs. This is particularly useful if you have a device (such as a server) that must belong to multiple VLANs. The device must have a NIC that supports 802.1Q tagging.

A single port can be a member of only one port-based VLAN. All additional VLAN membership for the port must be accompanied by tags. In addition to configuring the VLAN tag for the port, the server must have a *Network Interface Card (NIC)* that supports 802.1Q tagging.

Assigning a VLAN Tag

Each VLAN may be assigned an 802.1Q VLAN tag. As ports are added to a VLAN with an 802.1Q tag defined, you decide whether each port will use tagging for that VLAN. The default mode of the switch is to have all ports assigned to the VLAN named *default* with an 802.1Q VLAN tag (VLANid) of 1 assigned.

Not all ports in the VLAN must be tagged. As traffic from a port is forwarded out of the switch, the switch determines (in real time) if each destination port should use tagged or untagged packet formats for that VLAN. The switch adds and strips tags, as required, by the port configuration for that VLAN.

NOTE

Packets arriving tagged with a VLANid that is not configured on a port will be discarded.

Figure 4 illustrates the physical view of a network that uses tagged and untagged traffic.

Figure 4: Physical diagram of tagged and untagged traffic

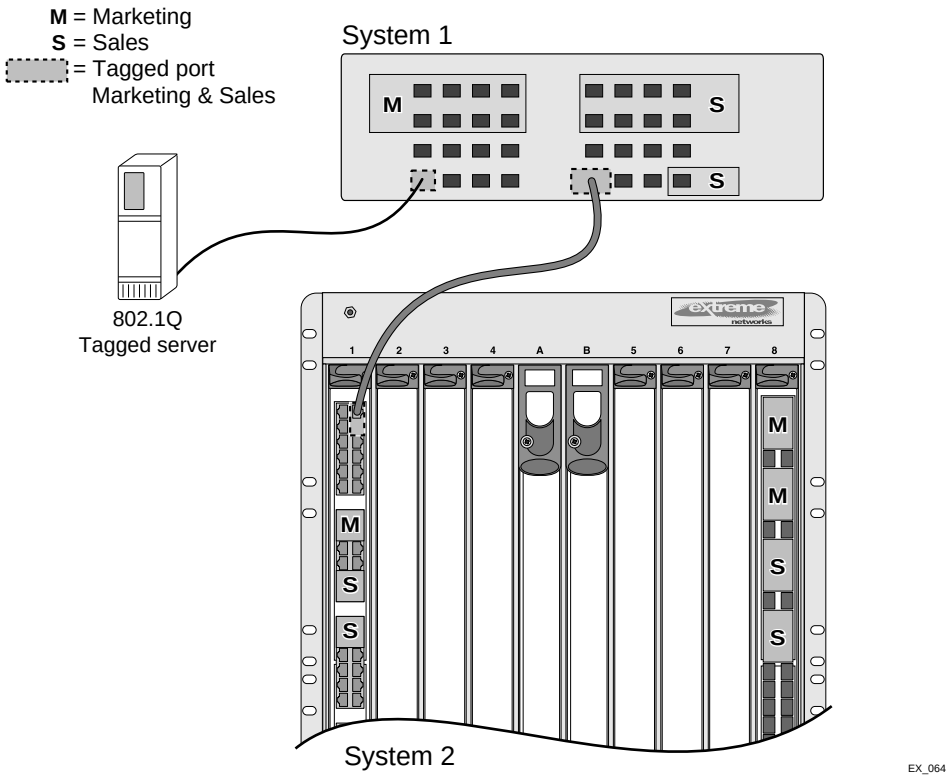
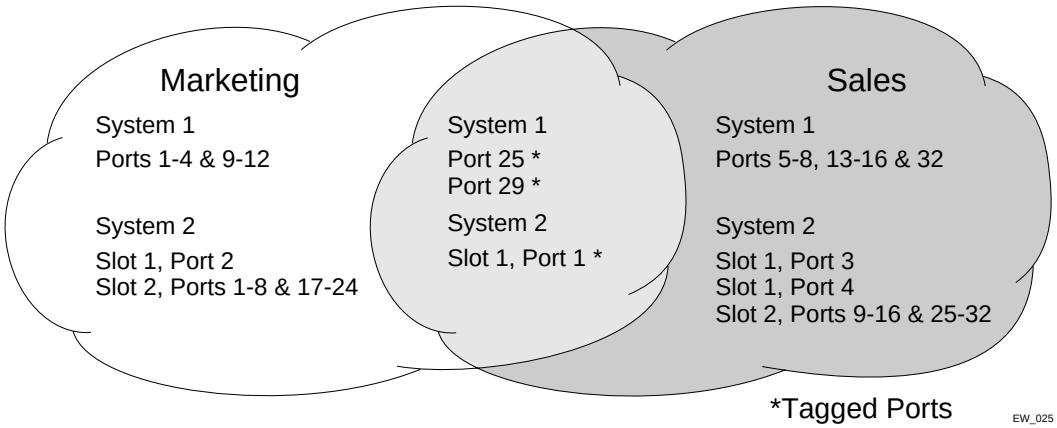


Figure 5 is a logical diagram of the same network.

Figure 5: Logical diagram of tagged and untagged traffic



In Figure 4 and Figure 5:

- The trunk port on each switch carries traffic for both VLAN *Marketing* and VLAN *Sales*.
- The trunk port on each switch is tagged.
- The server connected to port 25 on system 1 has a NIC that supports 802.1Q tagging.

- The server connected to port 25 on system 1 is a member of both VLAN *Marketing* and VLAN *Sales*.
- All other stations use untagged traffic.

As data passes out of the switch, the switch determines if the destination port requires the frames to be tagged or untagged. All traffic coming from and going to the server is tagged. Traffic coming from and going to the trunk ports is tagged. The traffic that comes from and goes to the other stations on this network is not tagged.

Mixing Port-Based and Tagged VLANs

You can configure the switch using a combination of port-based and tagged VLANs. A given port can be a member of multiple VLANs, with the stipulation that only one of its VLANs uses untagged traffic. In other words, a port can simultaneously be a member of one port-based VLAN and multiple tag-based VLANs.

NOTE

For the purposes of VLAN classification, packets arriving on a port with an 802.1Q tag containing a VLANid of zero are treated as untagged.

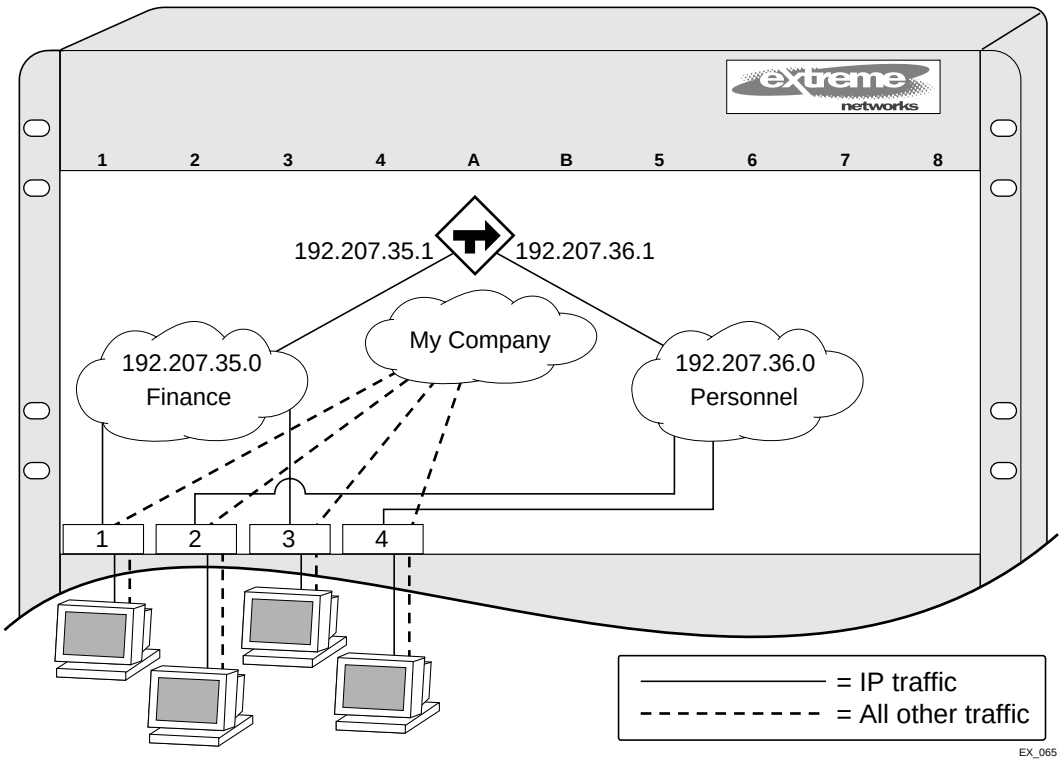
Protocol-Based VLANs

Protocol-based VLANs enable you to define a packet filter that the switch uses as the matching criteria to determine if a particular packet belongs to a particular VLAN.

Protocol-based VLANs are most often used in situations where network segments contain hosts running multiple protocols. For example, in Figure 6, the hosts are running both the IP and NetBIOS protocols.

The IP traffic has been divided into two IP subnets, 192.207.35.0 and 192.207.36.0. The subnets are internally routed by the switch. The subnets are assigned different VLAN names, *Finance* and *Personnel*, respectively. The remainder of the traffic belongs to the VLAN named *MyCompany*. All ports are members of the VLAN *MyCompany*.

Figure 6: Protocol-based VLANs



Predefined Protocol Filters

The following protocol filters are predefined on the switch:

- IP
- NetBIOS
- DECNet
- IPX_8022
- IPX_SNAP
- AppleTalk

Defining Protocol Filters

If necessary, you can define a customized protocol filter based on EtherType, Logical Link Control (LLC), and/or Subnetwork Access Protocol (SNAP). Up to six protocols may be part of a protocol filter. To define a protocol filter:

1 Create a protocol using the following command:

```
create protocol <name>
```

For example:

```
create protocol fred
```

The protocol name can have a maximum of 32 characters.

2 Configure the protocol using the following command:

```
configure protocol <name> add [etype | llc | snap] <hex> {[etype | llc | snap]
<hex>} ...
```

Supported protocol types include:

— `etype`—EtherType.

  The values for `etype` are four-digit hexadecimal numbers taken from a list maintained by the IEEE. This list can be found at the following URL:

  `http://standards.ieee.org/regauth/ethertype/index.html`

— `llc`—LLC Service Advertising Protocol (SAP).

  The values for `llc` are four-digit hexadecimal numbers that are created by concatenating a two-digit LLC Destination SAP (DSAP) and a two-digit LLC Source SAP (SSAP).

— `snap`—Ethertype inside an IEEE SNAP packet encapsulation.

  The values for `snap` are the same as the values for `etype`, described previously.

For example:

```
configure protocol fred add llc feff
configure protocol fred add snap 9999
```

A maximum of 15 protocol filters, each containing a maximum of six protocols, can be defined. On products that use the Inferno chip set, all 15 protocol filters can be active and configured for use. On all other platforms, no more than seven protocols can be active and configured for use.

NOTE

For more information on SNAP for Ethernet protocol types, see TR 11802-5:1997 (ISO/IEC) [ANSI/IEEE std. 802.1H, 1997 Edition].

Deleting a Protocol Filter

If a protocol filter is deleted from a VLAN, the VLAN is assigned a protocol filter of none. You can continue to configure the VLAN. However, no traffic is forwarded to the VLAN until a protocol is assigned to it.

Precedence of Tagged Packets Over Protocol Filters

If a VLAN is configured to accept tagged packets on a particular port, incoming packets that match the tag configuration take precedence over any protocol filters associated with the VLAN.

VLAN Names

Each VLAN is given a name that can be up to 32 characters. VLAN names can use standard alphanumeric characters. The following characters are not permitted in a VLAN name:

- Space
- Comma
- Quotation mark

VLAN names must begin with an alphabetical letter. The names can be no longer than 32 characters and must begin with an alphabetic character. The remainder of the name can be alphanumeric or contain underscore (_) characters. VLAN names cannot be keywords.

VLAN names can be specified using the tab key for command completion.

VLAN names are locally significant. That is, VLAN names used on one switch are only meaningful to that switch. If another switch is connected to it, the VLAN names have no significance to the other switch.

NOTE

You should use VLAN names consistently across your entire network.

Default VLAN

The switch ships with one default VLAN that has the following properties:

- The VLAN name is *default*.
- It contains all the ports on a new or initialized switch.
- The default VLAN is untagged on all ports. It has an internal VLANid of 1.

Renaming a VLAN

To rename an existing VLAN, use the following command:

```
configure vlan <vlan_name> name <new_name>
```

The following rules apply to renaming VLANs:

- After you change the name of the default VLAN, it cannot be changed back to *default*.
- You cannot create a new VLAN named *default*.
- You cannot change the VLAN name *MacVlanDiscover*. Although the switch accepts a name change, after it is rebooted, the original name is recreated.

Configuring VLANs on the Switch

This section describes the commands associated with setting up VLANs on the switch. Configuring a VLAN involves the following steps:

1. Create and name the VLAN.
2. Assign an IP address and mask (if applicable) to the VLAN, if needed.

NOTE

Each IP address and mask assigned to a VLAN must represent a unique IP subnet. You cannot configure the same IP subnet on different VLANs.

NOTE

If you plan to use this VLAN as a control VLAN for an EAPS domain, do NOT assign an IP address to the VLAN.

3. Assign a VLANid, if any ports in this VLAN will use a tag.
4. Assign one or more ports to the VLAN.

   As you add each port to the VLAN, decide if the port will use an 802.1Q tag.
5. For management VLAN, configure the default iproute for virtual router *VR-0*.

VLAN Configuration Examples

The following modular switch example creates a port-based VLAN named *accounting*, assigns the IP address 132.15.121.1, and assigns slot 2, ports 1, 2, 3, and 6, and slot 4, ports 1 and 2 to it:

```
create vlan accounting
configure accounting ipaddress 132.15.121.1
configure default delete port 2:1-2:3,2:6,4:1,4:2
configure accounting add port 2:1-2:3,2:6,4:1,4:2
```

NOTE

Because VLAN names are unique, you do not need to enter the keyword `vlan` *after you have created the unique VLAN name. You can use the VLAN name alone.*

The following stand-alone switch example creates a tag-based VLAN named *video*. It assigns the VLANid 1000. Ports 4 through 8 are added as tagged ports to the VLAN.

```
create vlan video
configure video tag 1000
configure video add port 4-8 tagged
```

The following stand-alone switch example creates a VLAN named *sales*, with the VLANid 120. The VLAN uses both tagged and untagged ports. Ports 1 through 3 are tagged, and ports 4 and 7 are untagged. Note that when not explicitly specified, ports are added as untagged.

```
create vlan sales
```

```
configure sales tag 120
configure sales add port 1-3 tagged
configure default delete port 4,7
configure sales add port 4,7
```

The following modular switch example creates a protocol-based VLAN named *ipsales*. Slot 5, ports 6 through 8, and slot 6, ports 1, 3, and 4-6 are assigned to the VLAN. In this example, you can add untagged ports to a new VLAN without first deleting them from the default VLAN, because the new VLAN uses a protocol other than the default protocol.

```
create vlan ipsales
configure ipsales protocol ip
configure ipsales add port 5:6-5:8,6:1,6:3-6:6
```

The following modular switch example defines a protocol filter, *myprotocol* and applies it to the VLAN named *myvlan*. This is an example only, and has no real-world application.

```
create protocol myprotocol
configure protocol myprotocol add etype 0xf0f0
configure protocol myprotocol add etype 0xffff
create vlan myvlan
configure myvlan protocol myprotocol
```

Displaying VLAN Settings

To display VLAN settings, use the following command:

```
show vlan {<vlan_name> | stpd}
```

The `show` command displays summary information about each VLAN, which includes:

- Name
- VLANid
- How the VLAN was created
- IP address
- STPD information
- Protocol information
- QoS profile information
- Ports assigned
- Tagged / untagged status for each port
- How the ports were added to the VLAN
- Number of VLANs configured on the switch

Use the `detail` option to display the detailed format.

Displaying Protocol Information

To display protocol information, use the following command:

```
show protocol {<name>}
```

This show command displays protocol information, which includes:

- Protocol name
- List of protocol fields
- VLANs that use the protocol

6 Forwarding Database (FDB)

This chapter describes the following topics:

- Overview of the FDB on page 75
- FDB Configuration Examples on page 77
- MAC-Based Security on page 78
- Displaying FDB Entries on page 78

Overview of the FDB

The switch maintains a database of all media access control (MAC) addresses received on all of its ports. It uses the information in this database to decide whether a frame should be forwarded or filtered.

FDB Contents

Each FDB entry consists of the MAC address of the device, an identifier for the port and VLAN on which it was received, and the age of the entry. Frames destined for MAC addresses that are not in the FDB are flooded to all members of the VLAN.

How FDB Entries Get Added

Entries are added into the FDB in the following ways:

- The switch can learn entries by examining packets it receives. The system updates its FDB with the source MAC address from a packet, the VLAN, and the port identifier on which the source packet is received.

  The ability to learn MAC addresses can be enabled or disabled on a port-by-port basis. You can also limit the number of addresses that can be learned, or you can “lock down” the current entries and prevent additional MAC address learning.
- You can enter and update entries using the command line interface (CLI).
- Certain static entries are added by the system upon switch boot up.

FDB Entry Types

FDB entries may be dynamic or static, and may be permanent or non-permanent. The following describes the types of entries that can exist in the FDB:

- **Dynamic entries**—A dynamic entry is learned by the switch by examining packets to determine the source MAC address, VLAN, and port information. The switch then creates or updates an FDB entry for that MAC address. Initially, all entries in the database are dynamic, except for certain entries created by the switch at boot up.

  Dynamic entries are flushed and relearned (updated) when any of the following take place:

  — A VLAN is deleted.

  — A VLAN identifier (VLANid) is changed.

  — A port mode is changed (tagged/untagged).

  — A port is deleted from a VLAN.

  — A port is disabled.

  — A port enters blocking state.

  — A port goes down (link down).

  A *non-permanent dynamic entry* is initially created when the switch identifies a new source MAC address that does not yet have an entry in the FDB. The entry may then be updated as the switch continues to encounter the address in the packets it examines. These entries are identified by the “d” flag in `show fdb` output.

  A *permanent dynamic entry* is created by command through the CLI, but may then be updated as the switch encounters the MAC address in the packets that it examines. A permanent dynamic entry is typically used to associate QoS profiles with the FDB entry. Permanent dynamic entries are identified by the “p” and “d” flags in `show fdb` output.

  Both types of dynamic entries age—a dynamic entry will be removed from the FDB (aged-out) if the device does not transmit for a specified period of time (the aging time). This prevents the database from becoming full with obsolete entries by ensuring that when a device is removed from the network, its entry is deleted from the database. The aging time is configurable. For more information about setting the aging time, see “Configuring the FDB Aging Time” on page 78.

- **Static entries**—A static entry does not age, and does not get updated through the learning process. It is maintained exactly as it was created. Conditions that cause dynamic entries to be updated, such as VLAN or port configuration changes, do not affect static entries.

  If the same MAC address is detected on another virtual port that is not defined in the static FDB entry for the MAC address, it is handled as a blackhole entry.

  A permanent static entry is created through the command line interface, and can be used to associate QoS profiles with a non-aging FDB entry. Permanent static entries are identified by the “s” and “p” flags in `show fdb` output.

Non-permanent static entries are created by the switch software for various reasons, typically upon switch boot up. They are identified by the "s" flag in `show fdb` output.

If the FDB entry aging time is set to zero, all entries in the database are considered static, non-aging entries. This means that they do not age, but they are still deleted if the switch is reset.

- **Permanent entries**—Permanent entries are retained in the database if the switch is reset or a power off/on cycle occurs. Permanent entries must be created by the system administrator through the command line interface. A permanent entry can either be a unicast or multicast MAC address.

  Permanent entries may be static, meaning they do not age or get updated, or they may be dynamic, meaning that they do age and can be updated via learning.

  Permanent entries can have QoS profiles associated with the MAC address. A different QoS profiles may be associated with the MAC address when it is a destination address (an egress QoS profile) than when it is a source address (ingress QoS profile).

  The stand-alone switches can support a maximum of 64 permanent entries, and the modular switches support a maximum of 254 permanent entries.

- **Blackhole entrie**s—A blackhole entry configures the switch to discard packets with a specified MAC address. Blackhole entries are useful as a security measure or in special circumstances where a specific source or destination address must be discarded. Blackhole entries may be created through the CLI, or they may be created by the switch when a port's learning limit has been exceeded.

  Blackhole entries are treated like permanent entries in the event of a switch reset or power off/on cycle. Blackhole entries are never aged out of the database.

Disabling MAC Address Learning

By default, MAC address learning is enabled on all ports. You can disable learning on specified ports using the following command:

```
disable learning port <port_list>
```

If MAC address learning is disabled, only broadcast traffic, EDP traffic, and packets destined to a permanent MAC address matching that port number, are forwarded. Use this command in a secure environment where access is granted via permanent forwarding databases (FDBs) per port.

FDB Configuration Examples

The following example adds a permanent static entry to the FDB:

```
create fdbentry 00:E0:2B:12:34:56 vlan marketing port 3:4
```

The permanent entry has the following characteristics:

- MAC address is 00:E0:2B:12:34:56.
- VLAN name is *marketing*.
- Slot number for this device is 3.
- Port number for this device is 4.

If the MAC address 00:E0:2B:12:34:56 is encountered on any port/VLAN other than VLAN *marketing*, port 3:4, it will be handled as a blackhole entry, and packets from that source will be dropped.

This example associates the QoS profile *qp2* with a dynamic entry for the device at MAC address 00:A0:23:12:34:56 on VLAN *net34* that will be learned by the FDB:

```
create fdbentry 00:A0:23:12:34:56 vlan net34 dynamic qosprofile qp2
```

This entry has the following characteristics:

- MAC address is 00:A0:23:12:34:56.
- VLAN name is *net34*.
- The entry will be learned dynamically.
- QoS profile *qp2* will be applied as an egress QoS profile when the entry is learned.

Overriding 802.1p Priority

This example associates the QoS profile *qp5* with the wildcard permanent FDB entry *any-mac* on VLAN v110:

```
create fdbentry any-mac vlan v110 dynamic ingress-qosprofile qp5
```

Configuring the FDB Aging Time

You can configure the again time for dynamic FDB entries using the following command:

```
configure fdb agingtime <seconds>
```

If the aging time is set to zero, all aging entries in the database are defined as static, nonaging entries. This means they will not age out, but non-permanent static entries can be deleted if the switch is reset. Supported aging is between 15 and 1,000,000 seconds.

MAC-Based Security

MAC-based security allows you to control the way the FDB is learned and populated. By managing entries in the FDB, you can block, assign priority (queues), and control packet flows on a per-address basis.

MAC-based security allows you to limit the number of dynamically-learned MAC addresses allowed per virtual port. You can also “lock” the FDB entries for a virtual port, so that the current entries will not change, and no additional addresses can be learned on the port.

You can also prioritize or stop packet flows based on the source MAC address of the ingress VLAN or the destination MAC address of the egress VLAN.

For detailed information about MAC-based security, see Chapter 9.

Displaying FDB Entries

To display FDB entries, use the following command:

```
show fdb {<mac_addr> | broadcast-mac | permanent | ports <portlist> | vlan
<vlan_name>}
```

where the following is true:

- `mac_address`—Displays the entry for a particular MAC address.

- `broadcast-mac`—Specifies the broadcast MAC address. May be used as an alternate to the colon-separated byte form of the address ff:ff:ff:ff:ff:ff
- `permanent`—Displays all permanent entries, including the ingress and egress QoS profiles.
- `ports <portlist>`—Displays the entries for a set of ports or slots and ports.
- `remap`—Displays the remapped FDB entries.
- `vlan <vlan name>`—Displays the entries for a VLAN.

With no options, the command displays all FDB entries.

See the *ExtremeWare XOS Command Reference Guide* for details of the commands related to the FDB.

7 Quality of Service (QoS)

This chapter covers the following topics:

- Overview of Policy-Based Quality of Service on page 82
- Applications and Types of QoS on page 82
- Configuring QoS on page 84
- QoS Profiles on page 84
- Traffic Groupings on page 85
  - Explicit Class of Service (802.1p and DiffServ) Traffic Groupings on page 86
  - Configuring DiffServ on page 87
  - Physical Groupings on page 89
- Verifying Configuration and Performance on page 89

Policy-based Quality of Service (QoS) is a feature of ExtremeWare XOS and the Extreme switch architecture that allows you to specify different service levels for traffic traversing the switch. Policy-based QoS is an effective control mechanism for networks that have heterogeneous traffic patterns. Using Policy-based QoS, you can specify the service level that a particular traffic type receives.

Overview of Policy-Based Quality of Service

Policy-based QoS allows you to protect bandwidth for important categories of applications or specifically limit the bandwidth associated with less critical traffic. For example, if voice–over-IP traffic requires a reserved amount of bandwidth to function properly, using policy-based QoS, you can reserve sufficient bandwidth critical to this type of application. Other applications deemed less critical can be limited so as to not consume excessive bandwidth. The switch contains separate hardware queues on every physical port. Each hardware queue is programmed by ExtremeWare XOS with bandwidth management and prioritization parameters. The bandwidth management and prioritization parameters that modify the forwarding behavior of the switch affect how the switch transmits traffic for a given hardware queue on a physical port.

The switch tracks and enforces the minimum and maximum percentage of bandwidth utilization transmitted on every hardware queue for every port. When two or more hardware queues on the same physical port are contending for transmission, the switch prioritizes bandwidth use so long as their respective bandwidth management parameters are satisfied. Up to eight physical queues per port are available.

NOTE

Policy-based QoS has no impact on switch performance. Using even the most complex traffic groupings has no cost in terms of switch performance.

Applications and Types of QoS

Different applications have different QoS requirements. The following applications are ones that you will most commonly encounter and need to prioritize:

- Voice applications
- Video applications
- Critical database applications
- Web browsing applications
- File server applications

General guidelines for each traffic type are given below and summarized in Table 11. Consider them as general guidelines and not strict recommendations. After QoS parameters have been set, you can monitor the performance of the application to determine if the actual behavior of the applications matches your expectations. It is very important to understand the needs and behavior of the particular applications you want to protect or limit. Behavioral aspects to consider include bandwidth needs, sensitivity to latency and jitter, and sensitivity and impact of packet loss.

Voice Applications

Voice applications typically demand small amounts of bandwidth. However, the bandwidth must be constant and predictable because voice applications are typically sensitive to latency (inter-packet delay) and jitter (variation in inter-packet delay). The most important QoS parameter to establish for voice applications is minimum bandwidth, followed by priority.

Video Applications

Video applications are similar in needs to voice applications, with the exception that bandwidth requirements are somewhat larger, depending on the encoding. It is important to understand the behavior of the video application being used. For example, in the playback of stored video streams, some applications can transmit large amounts of data for multiple streams in one “spike,” with the expectation that the end-stations will buffer significant amounts of video-stream data. This can present a problem to the network infrastructure, because it must be capable of buffering the transmitted spikes where there are speed differences (for example, going from Gigabit Ethernet to Fast Ethernet). Key QoS parameters for video applications include minimum bandwidth and priority.

Critical Database Applications

Database applications, such as those associated with ERP, typically do not demand significant bandwidth and are tolerant of delay. You can establish a minimum bandwidth using a priority less than that of delay-sensitive applications.

Web Browsing Applications

QoS needs for Web browsing applications cannot be generalized into a single category. For example, ERP applications that use a browser front-end may be more important than retrieving daily news information. Traffic groupings can typically be distinguished from each other by their server source and destinations. Most browser-based applications are distinguished by the dataflow being asymmetric (small dataflows from the browser client, large dataflows from the server to the browser client).

An exception to this may be created by some Java™ -based applications. In addition, Web-based applications are generally tolerant of latency, jitter, and some packet loss, however small packet-loss may have a large impact on perceived performance due to the nature of TCP. The relevant parameter for protecting browser applications is minimum bandwidth. The relevant parameter for preventing non-critical browser applications from overwhelming the network is maximum bandwidth. In addition, RED can be used to reduce session loss if the queue that floods Web traffic becomes over-subscribed.

File Server Applications

With some dependencies on the network operating system, file serving typically poses the greatest demand on bandwidth, although file server applications are very tolerant of latency, jitter, and some packet loss, depending on the network operating system and the use of TCP or UDP.

NOTE

Full-duplex links should be used when deploying policy-based QoS. Half-duplex operation on links can make delivery of guaranteed minimum bandwidth impossible.

Table 11 summarizes QoS guidelines for the different types of network traffic.

Table 11: Traffic Type and QoS Guidelines

Traffic Type	Key QoS Parameters
Voice	Minimum bandwidth, priority
Video	Minimum bandwidth, priority, buffering (varies)
Database	Minimum bandwidth

Table 11: Traffic Type and QoS Guidelines

Traffic Type	Key QoS Parameters
Web browsing	Minimum bandwidth for critical applications, maximum bandwidth for non-critical applications
File server	Minimum bandwidth

Configuring QoS

To configure QoS, you define how your switch responds to different categories of traffic by creating and configuring QoS profiles. You then group traffic into categories (according to application, as previously discussed) and assign each category to a QoS profile. Configuring QoS is a three-step process:

1. Configure the QoS profile.

   QoS profile—A class of service that is defined through minimum and maximum bandwidth parameters, and prioritization settings. The bandwidth and level of service that a particular type of traffic or traffic grouping receives is determined by assigning it to a QoS profile.

2. Create traffic groupings.

   Traffic grouping—A classification or traffic type that has one or more attributes in common. These can range from a physical port to IP layer 4 port information. You assign traffic groupings to QoS profiles to modify switch forwarding behavior. Traffic groupings transmitting out the same port that are assigned to a particular QoS profile share the assigned bandwidth and prioritization characteristics, and hence share the class of service.

3. Monitor the performance of the application with the QoS monitor to determine whether the policies are meeting the desired results.

The next sections describe each of these QoS components in detail.

QoS Profiles

A QoS profile defines a class of service by specifying traffic behavior attributes, such as bandwidth. The parameters that make up a QoS profile include:

- **Minimum bandwidth**—The minimum percentage of total link bandwidth that is reserved for use by a hardware queue on a physical port. Bandwidth unused by the queue can be used by other queues. The minimum bandwidth for all queues should add up to less than 100%. The default value on all minimum bandwidth parameters is 0%.
- **Maximum bandwidth**—The maximum percentage of total link bandwidth that can be transmitted by a hardware queue on a physical port. The default value on all maximum bandwidth parameters is 100%.
- **Priority**—The level of priority assigned to a hardware queue on a physical port. There are eight different available priority settings. By default, each of the default QoS profiles is assigned a unique priority. You would use prioritization when two or more hardware queues on the same physical port are contending for transmission on the same physical port, only after their respective bandwidth management parameters have been satisfied. If two hardware queues on the same physical port have the same priority, a round-robin algorithm is used for transmission, depending on the available link bandwidth.

— When configured to do so, the priority of a QoS profile can determine the 802.1p bits used in the priority field of a transmitted packet (described later).

— The priority of a QoS profile determines the DiffServ code point value used in an IP packet when the packet is transmitted (described later).

A QoS profile does not alter the behavior of the switch until it is assigned to a traffic grouping. Recall that QoS profiles are linked to hardware queues. There are multiple hardware queues per physical port. By default, a QoS profile links to the identical hardware queue across all the physical ports of the switch.

The default QoS profiles cannot be deleted. Also by default, a QoS profile maps directly to a specific hardware queue across all physical ports. The settings for the default QoS parameters are summarized in Table 12.

Table 12: QoS Parameters

Profile Name	Hardware Queue	Priority	Minimum Bandwidth	Maximum Bandwidth
Qp1	Q0	Low	0%	100%
Qp2	Q1	Lowhi	0%	100%
Qp3	Q2	Normal	0%	100%
Qp4	Q3	Normalhi	0%	100%
Qp5	Q4	Medium	0%	100%
Qp6	Q5	Mediumhi	0%	100%
Qp7	Q6	High	0%	100%
Qp8	Q7	Highhi	0%	100%

Traffic Groupings

After a QoS profile has been modified for bandwidth and priority, you assign traffic a grouping to the profile. A *traffic grouping* is a classification of traffic that has one or more attributes in common. Traffic is typically grouped based on the applications discussed starting on page 82.

Traffic groupings are separated into the following categories for discussion:

- Explicit packet class of service information, such as 802.1p or DiffServ (IP TOS)
- Physical configuration (physical source port association)

In the event that a given packet matches two or more grouping criteria, there is a predetermined precedence for which traffic grouping will apply. In general, the more specific traffic grouping takes precedence. By default, all traffic groupings are placed in the QoS profile Qp1. The supported traffic groupings are listed in Table 13. The groupings are listed in order of precedence (highest to lowest). The four types of traffic groupings are described in detail on the following pages.

Table 13: Traffic Groupings by Precedence

Explicit Packet Class of Service Groupings

Table 13: Traffic Groupings by Precedence (continued)

• DiffServ (IP TOS)
• 802.1P
Physical Groupings
• Source port

Explicit Class of Service (802.1p and DiffServ) Traffic Groupings

This category of traffic groupings describes what is sometimes referred to as *explicit packet marking,* and refers to information contained within a packet intended to explicitly determine a class of service. That information includes:

- IP DiffServ code points, formerly known as IP TOS bits
- Prioritization bits used in IEEE 802.1p packets

An advantage of explicit packet marking is that the class of service information can be carried throughout the network infrastructure, without repeating what can be complex traffic grouping policies at each switch location. Another advantage is that end stations can perform their own packet marking on an application-specific basis. Extreme switch products have the capability of observing packet marking information with no performance penalty.

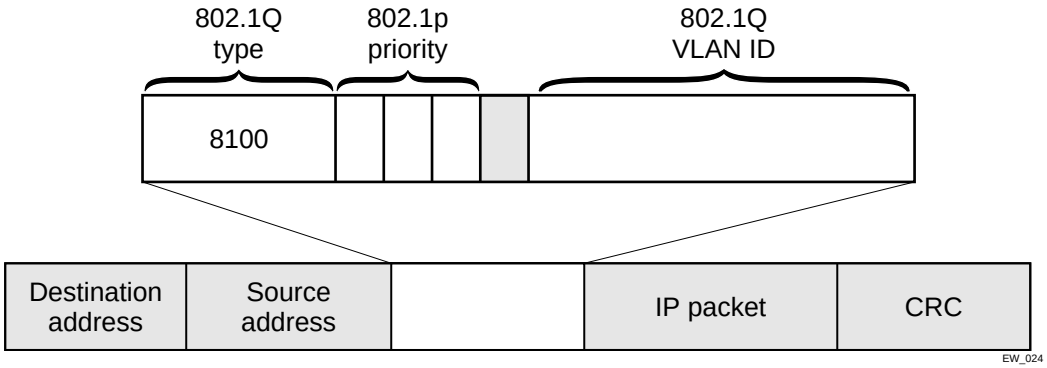
The documented capabilities for 802.1p priority markings or DiffServ capabilities (if supported) are not impacted by the switching or routing configuration of the switch. For example, 802.1p information can be preserved across a routed switch boundary and DiffServ code points can be observed across a layer 2 switch boundary.

Configuring 802.1p Priority

Extreme switches support the standard 802.1p priority bits that are part of a tagged Ethernet packet. The 802.1p bits can be used to prioritize the packet, and assign it to a particular QoS profile.

When a packet arrives at the switch, the switch examines the 802.1p priority field maps it to a specific hardware queue when subsequently transmitting the packet. The 802.1p priority field is located directly following the 802.1Q type field, and preceding the 802.1Q VLAN ID, as shown in Figure 7.

Figure 7: Ethernet packet encapsulation



Observing 802.1p Information

When ingress traffic that contains 802.1p prioritization information is detected by the switch, the traffic is mapped to various hardware queues on the egress port of the switch. Eight hardware queues are supported. The transmitting hardware queue determines the bandwidth management and priority characteristics used when transmitting packets.

To control the mapping of 802.1p prioritization values to hardware queues, 802.1p prioritization values can be mapped to a QoS profile. The default mapping of each 802.1p priority value to QoS profile is shown in Table 14.

Table 14: 802.1p Priority Value-to-QoS Profile Default Mapping

Priority Value	QoS Profile
0	Qp1
1	Qp2
2	Qp3
3	Qp4
4	Qp5
5	Qp6
6	Qp7
7	Qp8

Changing the Default 802.1p Mapping

By default, a QoS profile is mapped to a hardware queue, and each QoS profile has configurable bandwidth parameters and priority. In this way, an 802.1p priority value seen on ingress can be mapped to a particular QoS profile and with specific bandwidth management and priority behavior.

To change the default mappings of QoS profiles to 802.1p priority values, use the following command:

```
configure dot1p type <dot1p_priority> qosprofile <qosprofile>
```

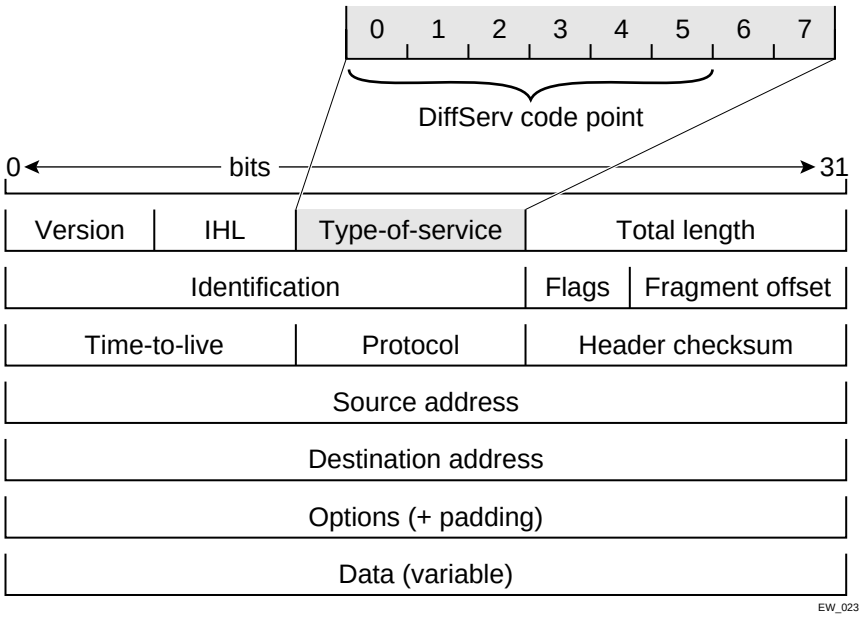
Configuring DiffServ

Contained in the header of every IP packet is a field for IP Type of Service (TOS), now also called the DiffServ field. The TOS field is used by the switch to determine the type of service provided to the packet.

Observing DiffServ code points as a traffic grouping mechanism for defining QoS policies and overwriting the Diffserv code point fields are supported.

Figure 8 shows the encapsulation of an IP packet header.

Figure 8: IP packet header encapsulation



Observing DiffServ Information

When a packet arrives at the switch on an ingress port, the switch examines the first six of eight TOS bits, called the *code point*. The switch can assign the QoS profile used to subsequently transmit the packet based on the code point. The QoS profile controls a hardware queue used when transmitting the packet out of the switch, and determines the forwarding characteristics of a particular code point. Viewing DiffServ information can be enabled or disabled; by default it is disabled. To view DiffServ information, use the following command:

```
show diffserv
```

Changing DiffServ Code point assignments in the Q0S Profile

Because the code point uses six bits, it has 64 possible values ($2^6 = 64$). Be default, the values are grouped and assigned to the default QoS profiles listed in Table 15.

Table 15: Default Code Point-to-QoS Profile Mapping

Code Point	QoS Profile
0-7	Qp1
8-15	Qp2
16-23	Qp3
24-31	Qp4
32-39	Qp5
40-47	Qp6
48-55	Qp7
56-63	Qp8

You can change the QoS profile assignment for all 64 code points using the following command:

```
configure diffserv examination code-point <code-point> qosprofile <qosprofile>
```

Once assigned, the rest of the switches in the network prioritize the packet using the characteristics specified by the QoS profile.

To verify the DiffServ configuration, use the following command:

```
show ports {<port_list>} qosmonitor
```

DiffServ Example

In this example, we use DiffServ to signal a class of service throughput and assign any traffic coming from network 10.1.2.x with a specific DiffServ code point. This allows all other network switches to send and observe the Diffserv code point instead of repeating the same QoS configuration on every network switch.

To configure the switch that handles incoming traffic from network 10.1.2.x, follow these steps:

1. Configure parameters of the QoS profile QP3:

   ```
   configure qp3 min 10 max 100
   ```

2. Configure the switch so that other switches can signal class of service that this switch should observe:

   ```
   enable diffserv examination
   ```

Physical Groupings

A source port traffic grouping implies that any traffic sourced from this physical port uses the indicated QoS profile when the traffic is transmitted out to any other port. To configure a source port traffic grouping, use the following command:

```
configure ports <port_list> qosprofile <qosprofile>
```

In the following modular switch example, all traffic sourced from slot 5 port 7 uses the QoS profile named *qp3* when being transmitted.

```
configure ports 5:7 qosprofile qp3
```

Verifying Physical Groupings

To verify settings on ports, use the following command:

```
show ports {<port_list>} information {detail}
```

The same information is also available for ports using the following command:

```
show ports {<port_list>} qosmonitor
```

Verifying Configuration and Performance

After you have created QoS policies that manage the traffic through the switch, you can use the QoS monitor to determine whether the application performance meets your expectations.

QoS Monitor

The QoS monitor is a utility that monitors the hardware queues associated with any port(s). The QoS monitor keeps track of the number of frames that a specific queue is responsible for transmitting on a physical port.

Real-Time Performance Monitoring

QoS features real-time performance monitoring with a snapshot display of the monitored ports. To view real-time switch per-port performance, use the following command:

```
show ports {<port_list>} qosmonitor
```

Displaying QoS Profile Information

The QoS monitor can also be used to verify the QoS configuration and monitor the use of the QoS policies that are in place. To display QoS information on the switch, use the following command:

```
show qosprofile {<qosprofile>}
```

Displayed information includes:

- QoS profile name
- Minimum bandwidth
- Maximum bandwidth
- Priority

Additionally, QoS information can be displayed from the traffic grouping perspective by using one or more of the following command, which displays QoS information from the port:.

```
show ports {<port_list>} information {detail}
```

8 Status Monitoring and Statistics

This chapter describes the following topics:

- Status Monitoring on page 91
- Slot Diagnostics on page 91
- Port Statistics on page 93
- Port Errors on page 94
- Port Monitoring Display Keys on page 95
- System Temperature on page 95
- System Health Checking on page 96
- System Redundancy on page 96
- Event Management System/Logging on page 100

Viewing statistics on a regular basis allows you to see how well your network is performing. If you keep simple daily records, you will see trends emerging and notice problems arising before they cause major network faults. In this way, statistics can help you get the best out of your network.

Status Monitoring

The status monitoring facility provides information about the switch. This information may be useful for your technical support representative if you have a problem. ExtremeWare XOS includes many show commands that display information about different switch functions and facilities.

NOTE

For more information about show commands for a specific ExtremeWare XOS feature, see the appropriate chapter in this guide.

Slot Diagnostics

The BlackDiamond switch provides a facility for running normal or extended diagnostics on an I/O module or a Management Switch Fabric Module (MSM) without affecting the operation of the rest of the system.

If you run the diagnostic routine on an I/O module, that module is taken offline while the diagnostic test is performed. Traffic to and from the ports on the module are temporarily unavailable. Once the diagnostic test is completed, the I/O module is reset and becomes operational again.

You must enter the Bootloader to run the diagnostic routine on the MSM module. The module is taken offline while the diagnostics test is performed. When the diagnostic test is complete, the MSM reboots, and becomes operational again.

Running Diagnostics on I/O Modules

To manually run diagnostics on I/O modules, use the following command:

```
run diagnostics [extended | normal] slot <slot>
```

where the following is true:

- `normal`—Takes the switch fabric and ports offline, and performs a simple ASIC and packet loopback test on all ports. The test is completed in 30 seconds. CPU and out-of-band management ports are not tested in this mode. As a result, console and Telnet access from the management port is available during this routine.
- `extended`—Takes the switch fabric and ports offline, and performs extensive ASIC, ASIC-memory, and packet loopback tests. Extended diagnostic tests take a maximum of 15 minutes. The CPU is not tested. Console access is available during extended diagnostics.
- `<slot>`—Specifies the slot number of an I/O module. When the diagnostics test is complete, the system attempts to bring the I/O module back online.

Running Diagnostics on MSM Modules

To manually run diagnostics on an MSM module, you must first enter the Bootloader and then issue a series of commands in the Bootloader.

To access the Bootloader:

1. Attach a serial cable to the console port of the switch.
2. Attach the other end of the serial cable to a properly configured terminal or terminal emulator, power cycle the switch and depress any ASCII key on the keyboard of the terminal during the boot up process.

   NOTE

   To access the Bootloader, you can depress any key until the applications load and run on the switch.

   As soon as you see the `BOOTLOADER->` prompt, release the key. From here, you can run the diagnostics on the MSM.

To run diagnostics on the MSM:

1. Identify the images currently running by using the `show images` command.
2. Run diagnostics on the MSM by using the following command:

   ```
   boot [1-4]
   ```

   The numbers 1 through 4 correlate to specific images and diagnostics on the MSM:

   - 1—XOS primary image

- 2—XOS secondary image
- 3—Diagnostics for image 1 (initiates diagnostics for the primary image)
- 4—Diagnostics for image 2 (initiates diagnostics for the secondary image)

For example, to run diagnostics on the primary image, use the following command:

```
boot 3
```

When the test is finished, the MSM reboots and runs the XOS software.

Viewing Slot Diagnostics

To display the status of the last diagnostic test run on the switch, use the following command:

```
show diagnostics [msm-a | msm-b | slot <slot>]
```

Port Statistics

ExtremeWare XOS provides a facility for viewing port statistic information. The summary information lists values for the current counter against each port on each operational module in the system, and it is refreshed approximately every 2 seconds. Values are displayed to nine digits of accuracy.

To view port statistics, use the following command:

```
show ports <port_list> statistics
```

The switch collects the following port statistic information:

- **Link Status—**The current status of the link. Options are:
  - Ready (the port is ready to accept a link).
  - Active (the link is present at this port).
- **Transmitted Packet Count (Tx Pkt Count)—**The number of packets that have been successfully transmitted by the port.
- **Transmitted Byte Count (Tx Byte Count)—**The total number of data bytes successfully transmitted by the port.
- **Received Packet Count (Rx Pkt Count)—**The total number of good packets that have been received by the port.
- **Received Byte Count (RX Byte Count)—**The total number of bytes that were received by the port, including bad or lost frames. This number includes bytes contained in the Frame Check Sequence (FCS), but excludes bytes in the preamble.
- **Received Broadcast (RX Bcast)—**The total number of frames received by the port that are addressed to a broadcast address.
- **Received Multicast (RX Mcast)—**The total number of frames received by the port that are addressed to a multicast address.

Port Errors

The switch keeps track of errors for each port.

To view port transmit errors, use the following command:

```
show ports {<port_list>} txerrors
```

The switch collects the following port transmit error information:

- **Port Number**
- **Link Status—**The current status of the link. Options are:
  - Ready (the port is ready to accept a link).
  - Active (the link is present at this port).
- **Transmit Collisions (TX Coll)—**The total number of collisions seen by the port, regardless of whether a device connected to the port participated in any of the collisions.
- **Transmit Late Collisions (TX Late Coll)—**The total number of collisions that have occurred after the port's transmit window has expired.
- **Transmit Deferred Frames (TX Deferred)—**The total number of frames that were transmitted by the port after the first transmission attempt was deferred by other network traffic.
- **Transmit Errored Frames (TX Error)—**The total number of frames that were not completely transmitted by the port because of network errors (such as late collisions or excessive collisions).
- **Transmit Parity Frames (TX Parity)**—The bit summation has a parity mismatch.

To view port receive errors, use the following command:

```
show ports {<port_list>} rxerrors
```

The switch collects the following port receive error information:

- **Receive Bad CRC Frames (RX CRC)—**The total number of frames received by the port that were of the correct length, but contained a bad FCS value.
- **Receive Oversize Frames (RX Over)—**The total number of good frames received by the port greater than the supported maximum length of 1,522 bytes.
- **Receive Undersize Frames (RX Under)—**The total number of frames received by the port that were less than 64 bytes long.
- **Receive Fragmented Frames (RX Frag)—**The total number of frames received by the port were of incorrect length and contained a bad FCS value.
- **Receive Jabber Frames (RX Jab)—**The total number of frames received by the port that was of greater than the support maximum length and had a Cyclic Redundancy Check (CRC) error.
- **Receive Alignment Errors (RX Align)—**The total number of frames received by the port that occurs if a frame has a CRC error and does not contain an integral number of octets.
- **Receive Frames Lost (RX Lost)—**The total number of frames received by the port that were lost because of buffer overflow in the switch.

Port Monitoring Display Keys

Table 16 describes the keys used to control the displays that appear when you issue any of the `show port` commands.

Table 16: Port monitoring display keys

Key(s)	Description
U	Displays the previous page of ports.
D	Displays the next page of ports.
[Esc] or [Return]	Exits from the screen.
0	Clears all counters.
[Space]	Cycles through the following screens: • Packets per second • Bytes per second • Percentage of bandwidth Available using the `show port utilization` command only.

System Temperature

You can view the temperature of the system and the I/O and management modules, in celsius, for the BlackDiamond 10808 chassis.

To view the system temperature, use the following command:

```
show temperature
```

The following is sample output from this command:

```
Field Replaceable Units    Temp (C)
-----------------------    --------
Slot-1  : PCSimC2             20.10
Slot-2  : PCSimC2             20.20
Slot-3  : PCSimC2             20.30
Slot-4  : PCSimC2             20.40
Slot-5  : PCSimC256           20.50
Slot-6  : PCSimC256           20.60
Slot-7  : PCSimC256           20.70
Slot-8  :
MSM-A   : PCSimMSM            20.90
MSM-B   : PCSimMSM            21.00
```

You can also view the temperature of the power supplies and the fan tray. All temperatures are displayed in celsius.

To view the current temperature of the power supplies, use the following command:

```
show powersupplies {detail}
```

The following sample output displays the temperature information:

```
PowerSupply 1 information:
 ...
 Temperature:    30.1 deg C
 ...
```

To view the current temperature of the fan trays, use the following command:

```
show fans {detail}
```

The following sample output displays the temperature information:

```
FanTray 1 information:
 ...
 Temperature:    25.1 deg C
 ...
```

System Health Checking

The system health checker tests the backplane, the CPU, and I/O modules by periodically forwarding packets and checking for the validity of the forwarded packets.

To enable the system health checker, use the following command:

```
enable sys-health-check slot <slot>
```

To disable the system health checker, use the following command:

```
disable sys-health-check slot <slot>
```

To configure the how often packets are forwarded, use the following command:

```
configure sys-health-check interval <interval>
```

System Redundancy

If you install two MSMs in the chassis, one assumes the role of primary (master) and the other assumes the role of standby (backup). The primary MSM provides all of the switch management functions including bringing up and programming the I/O modules, running the bridging and routing protocols, and configuring the switch. The primary also keeps in sync with the standby MSM in case the standby needs to take over the primary role if the primary fails.

Table 17 describes the terms associated with system redundancy.

Table 17: System redundancy terms

Term	Description
Node	A node is a CPU that runs the XOS management applications on the switch. Each MSM installed in the chassis is a node.

Table 17: System redundancy terms (continued)

Term	Description
Node Manager	The Node Manager is a process that performs leader election between multiple nodes in the system. In simple terms, the Node Manager elects the primary and backup MSMs. If there is only one MSM installed, that MSM becomes the primary MSM. To determine the primary node, the Node Manager sends hello messages to all of the nodes in the system, computes the health of the system, and checks the parameters of each node (for example the node state, configuration, priority, etc.).
Device Manager	The Device Manager is a process that runs on every node and is responsible for monitoring and controlling all of the devices in the system. The Device Manager consists of a process and a client library that is dynamically linked to every process that runs under XOS. The library manages the communication of data between the device manager server and the library so that all common system specific data required by this process is stored and available to it immediate from it own private memory.
Node Election	Node election is the actual process of electing the primary and backup node. Once elected, the primary node coordinates all system activities between the nodes and becomes the main computation and management point of contact. The backup node facilitates a faster recovery if the primary node fails.
Checkpointing	The process of copying the active state configurations from the primary MSM to the backup MSM.

The following sections describe the process, configuration, and monitoring of system redundancy.

Electing the Node

The node election process is a connectionless, un-reliable messaging mechanism between the nodes installed in the chassis. The health of the node is also important in selecting the primary node. The Device Manager collects the node health information and forwards that information to the Node Manager. The Node Manager then computes the quality of the node which is later used in leader election.

When two nodes exchange their health information, they come to a conclusion as to which is the healthier node. Based on the election results obtained from all of the nodes, the healthiest node wins the election criteria.

At the end of the election process, a primary node is selected along with a backup or secondary node. The primary node runs the switch management functions, and the backup node is available to run the switch management functions if the primary fails.

The parameters used to determine the primary node are:

- Node state—The node state must be ONLINE to participate in leader election and to be selected primary. If the node is in the INIT, OFFLINE, or FAIL states, the node will not participate in leader election. For more information about the node states, see “Viewing Node Statistics” on page 100.
- Configuration priority—User assigned priority. The configured priority is compared only after the node meets the minimum thresholds in each category for it to be healthy.
- Control channel bandwidth—This is a function of the number of links available and the total bandwidth of these links.
- Software health—This number represents the percent of processes available.
- Software version—Represents the software version the node is running.
- Health of secondary hardware components—Represents the health of the power supplies, fans, etc.

- Slot ID—The number of the slot where the node is installed.
- MAC address—The MAC address is used to determine the primary node if all other parameters are equal.

Configuring Node Parameters

To configure the parameters of a node, use the following command:

```
configure node slot <slot_id> priority <node_pri>
```

Configuring the Node State

You can bring a node offline to run diagnostics or perform software upgrades. If you specify the primary node to be offline, the system will failover to the backup node and the previous primary node will become the new backup node. If you specify the backup node to be offline, the processes on the primary will stop checkpointing because the backup node is unavailable.

NOTE

If you configure the node to be offline, it is not available to participate in leader election.

To bring a node offline, use the following command:

```
configure node slot <slot_id> offline priority <node_pri>
```

To bring a node back online, use the following command:

```
configure node slot <slot_id> online priority <node_pri>
```

Relinquishing Primary Status

You can force the primary node to failover to the backup thereby relinquishing its primary status. You execute this command on the primary node.

To failover to the backup node, use the following command:

```
failover {force}
```

If you specify `force`, the primary node will failover provided the backup node can take over as primary. If there is no backup node, the primary will transition to the standby state and a new election will start based on the current health of the node and a new primary will take over.

If you do not specify `force`, failover will not occur unless the backup node (MSM) is in sync with the primary.

Replicating Data Between the Nodes

ExtremeWare XOS replicates configuration information between the primary MSM to the backup MSM so the system can recover if the primary MSM fails. This method of replicating data is known as checkpointing. Checkpointing is the process of copying the active state configurations from the primary MSM to the backup MSM.

Replicating data consists of the following three steps:

1. Relaying configuration information from the master to the backup
2. Ensuring that each individual process running on the system is in sync with the backup
3. Checkpointing any new state changes from the master to the backup

Relaying Configuration Information

This is the first level of checkpointing that is required to facilitate a failover: the primary's configuration information is transferred to the backup MSM, and the backup MSM ignores their own flash configuration file.

When you initially boot the switch, the primary MSM configuration takes effect. During any standby initialization, the primary's saved configuration is copied to the local flash, and the current active configuration is transferred to the backup processes. As you make configuration changes to the primary MSM, they are relayed to the backup and incorporated into its configuration copy.

NOTE

To ensure that all of the configuration commands in the backup's flash is updated, issue the `save` *command after you make any changes.*

If a failover occurs, the backup MSM continues to use the primary's active configuration. If the backup determines that it does not have the primary's active configuration, it will use the configuration stored in its flash memory.

NOTE

If you issue the `reboot` *command before you save your configuration changes, the switch prompts you to save your changes.*

Synchronizing the Backup

The next step in checkpointing requires that the primary and backup configurations be in sync. Since ExtremeWare XOS runs a series of processes (for example the device manager and the node manager), a process starts checkpointing only after all the processes that it depends on have transferred their states to the backup MSM.

After a process completes it checkpoints, this triggers the next process to proceed with its checkpointing. You can also view the progress of the checkpointing. See the section "Viewing Checkpoint Statistics" later in this chapter.

Relaying New State Information

The final step in checkpointing requires that any new configuration information or state changes that occur on the master be immediately relayed to the backup. This ensures that the backup has the most up to date and accurate configuration information.

Viewing Checkpoint Statistics

As previously noted, checkpointing is the process of copying the active state configurations from the primary MSM to the backup MSM. Use the following command to view and check the status of one or more processes being copied from the master to the backup MSM:

```
show checkpoint-data {<process>}
```

This command displays, in percentages, the amount of copying completed by each process and the traffic statistics between the process on both the primary and the backup MSMs.

Viewing Node Statistics

ExtremeWare XOS allows you to view node statistic information. Each node installed in your system is self-sufficient and runs the XOS management applications. By reviewing this output, you can see the general health of the system along with other node parameters.

To view the node statistics information, use the following command:

```
show node {detail}
```

Table 18 lists the node statistic information collected by the switch.

Table 18: Node states

Node State	Description
INIT	The initial state where the node is being initialized. A node stays in this state when it is coming up and remains in this state until it has been fully initialized. Being fully initialized means that all of the hardware has been initialized correctly and there are no diagnostic faults.
OFFLINE	You have requested the node to go down. Use this mode to run diagnostics or perform software upgrades. In this mode, the node is not available to participate in leader election.
FAIL	The node has failed and needs to be restarted or repaired. The node reaches this state if the system has a hardware or software failure.
MASTER	This node is the primary node and is responsible for all of the switch management functions.
BACKUP	This node is the designated backup (secondary) node and will be used to failover if the primary is unavailable. This node will become the primary node. This node also receives the checkpoints from the primary.
STANDBY	This node is in the standby state.If the primary is not available, this node will enter leader election and transition to primary if it wins. If you request a node to enter the backup state, it will enter the standby state before entering the backup state.

Event Management System/Logging

We use the general term, event, for any type of occurrence on a switch which could generate a log message, or require an action. For example, a link going down, a user logging in, a command entered on the command line, or the software executing a debugging statement, are all events that might generate a log message. The system for saving, displaying, and filtering events is called the Event Management System (EMS). With EMS, you have many options about which events generate log messages, where the messages are sent, and how they are displayed. Using EMS you can:

- Send event messages to a number of logging targets (for example, syslog host, and NVRAM)
- Filter events on a per-target basis, by:
  - Component, subcomponent, or specific condition (for example, BGP messages, *IGMP.Snooping* messages, or the *IP.Forwarding.SlowPathDrop* condition)
  - Match expression (for example, any messages containing the string “user5”)

  - Matching parameters (for example, only messages with source IP addresses in the 10.1.2.0/24 subnet)
  - Severity level (for example, only messages of severity critical, error, or warning)
- Change the format of event messages (for example, display the date as “12-May-2003” or “2003-05-12”)
- Display log messages in real-time, and filter the messages that are displayed, both on the console and from telnet sessions
- Display stored log messages from the memory buffer or NVRAM
- Upload event logs stored in memory buffer or NVRAM to a TFTP server
- Display counts of event occurrences, even those not included in filter
- Display debug information, using a consistent configuration method

Sending Event Messages to Log Targets

Five types of targets can receive log messages:

- Console display
- Current session (telnet or console display)
- Memory buffer (can contain 200-20,000 messages)
- NVRAM (messages remain after reboot)
- Syslog host

The first four types of targets exist by default, but before enabling any syslog host, the host’s information needs to be added to the switch using the `configure syslog` command. Extreme Networks EPICenter can be a syslog target.

By default, the memory buffer and NVRAM targets are already enabled and receive messages. To start sending messages to the targets, use the following command:

```
enable log target [console | memory-buffer | nvram | session | syslog [all |
ipaddress] [local0 ... local7]]]
```

After it has been enabled, the target receives the messages it is configured for. See “Target Configuration” later in this chapter for information on viewing the current configuration of a target. The memory buffer can only contain the configured number of messages, so the oldest message is lost when a new message arrives, and the buffer is full.

Use the following command to stop sending messages to the target:

```
disable log target [console | memory-buffer | nvram | session | syslog [all |
<ipaddress> ] [local0 ... local7]]]
```

NOTE

Refer to your UNIX documentation for more information about the syslog host facility.

Filtering Events Sent to Targets

Not all event messages are sent to every enabled target. Each target receives only the messages that it is configured for.

Target Configuration

To specify the messages to send to an enabled target, you will set a message severity level, a filter name, and a match expression. These items determine which messages are sent to the target. You can also configure the format of the messages in the targets. For example, the console display target is configured to get messages of severity `info` and greater, the NVRAM target gets messages of severity `warning` and greater, and the memory buffer target gets messages of severity `debug-data` and greater. All the targets are associated by default with a filter named *DefaultFilter*, that passes all events at or above the default severity threshold. All the targets are also associated with a default match expression that matches any messages (the expression that matches any message is displayed as `Match : (none)` from the command line). And finally, each target has a format associated with it.

To display the current log configuration of the targets, use the following command:

```
show log configuration target {console | memory-buffer | nvram | session | syslog
<ipaddress> [local0 ... local7]}
```

To configure a target, there are specific commands for filters, formats, and severity that are discussed in the following sections.

Severity

Messages are issued with one of the severity level specified by the standard BSD syslog values (RFC 3164), `critical`, `error`, `warning`, `notice`, and `info`, plus three severity levels for extended debugging, `debug-summary`, `debug-verbose`, and `debug-data`. Note that RFC 3164 syslog values `emergency` and `alert` are not needed since `critical` is the most severe event in the system.

The three severity levels for extended debugging, `debug-summary`, `debug-verbose`, and `debug-data`, require that debug mode be enabled (which may cause a performance degradation). See “Displaying Debug Information” on page 111 for more information about debugging.

Table 19: Severity levels assigned by the switch

Level	Description
Critical	A serious problem has been detected which is compromising the operation of the system and that the system cannot function as expected unless the situation is remedied. The switch may need to be reset.
Error	A problem has been detected which is interfering with the normal operation of the system and that the system is not functioning as expected.
Warning	An abnormal condition, not interfering with the normal operation of the system, has been detected which may indicate that the system or the network in general may not be functioning as expected.
Notice	A normal but significant condition has been detected, which signals that the system is functioning as expected.
Info (Informational)	A normal but potentially interesting condition has been detected, which signals that the system is functioning as expected and simply provides potentially detailed information or confirmation.
Debug-Summary	A condition has been detected that may interest a developer determining the reason underlying some system behavior.

Table 19: Severity levels assigned by the switch (continued)

Level	Description
Debug-Verbose	A condition has been detected that may interest a developer analyzing some system behavior at a more verbose level than provided by the debug summary information.
Debug-Data	A condition has been detected that may interest a developer inspecting the data underlying some system behavior.

To configure the severity level of the messages sent to a target, there is more than one command that you can use. The most direct way to set the severity level of all the sent messages is to use the following command:

```
configure log target [console | memory-buffer | nvram | session | syslog [<all |
ipaddress> [local0 ... local7]]] {severity <severity> {only}}
```

When you specify a severity level, messages of that severity and greater will be sent to the target. If you want only messages of the specified severity to be sent to the target, use the keyword `only`. For example, specifying `severity warning` will send warning, error, and critical messages, but specifying `severity warning only` will just send warning messages.

Another command that can be used to configure severity levels is the command used to associate a filter with a target:

```
configure log target [console | memory-buffer | nvram | session | syslog [all |
<ipaddress> [local0 ... local7]]] filter <filter-name> {severity <severity> {only}}
```

When you specify a severity level as you associate a filter with a target, you further restrict the messages reaching the target. The filter may only allow certain categories of messages to pass. Only the messages that pass the filter, and then pass the specified severity level will reach the target.

Finally, you can specify the severity levels of messages that reach the target by associating a filter with a target. The filter can specify exactly which message it will pass. Constructing a filter is described in “Filtering By Components and Conditions” on page 105.

Components and Conditions

The event conditions detected by ExtremeWare XOS are organized into components and subcomponents. To get a listing of the components and subcomponents in your release of ExtremeWare XOS, use the following command:

```
show log components {<event component> | version}
```

For example, to get a listing of the components and subcomponents in your system, use the following command:

```
show log components
```

The partial output produced by the command is similar to the following:

```
* BD-PC.5 # show log components
                                                                  Severity
Component           Title                                         Threshold
------------------- --------------------------------------------- -------------
...
...
STP                 Spanning-Tree Protocol (STP)                  Error
```

```
        InBPDU      STP In BPDU subcomponent                        Warning
        OutBPDU     STP Out BPDU subcomponent                       Warning
        System      STP System subcomponent                         Error
...
...
```

The display above lists the components, subcomponents, and the default severity threshold assigned to them. A period (.) is used to separate component, subcomponent, and condition names in EMS. For example, you can refer to the *InBPDU* subcomponent of the STP component as *STP.InBPDU*. On the CLI, you can abbreviate or TAB complete any of these.

A component or subcomponent will often have several conditions associated with it. To see the conditions associated with a component, use the following command:

```
show log events [<event condition> | [all | <event component>] {severity <severity>
{only}}] {details}
```

For example, to see the conditions associated with the *STP.InBPDU* subcomponent, use the following command:

```
show log events stp.inbpdu
```

The output produced by the command is similar to the following:

```
Comp    SubComp     Condition               Severity      Parameters
------- ----------- ----------------------- ------------- ----------
STP     InBPDU      Drop                    Error          2 total
STP     InBPDU      Dump                    Debug-Data     3 total
STP     InBPDU      Trace                   Debug-Verbose  2 total
STP     InBPDU      Ign                     Debug-Summary  2 total
STP     InBPDU      Mismatch                Warning        2 total
```

The display above lists the five conditions contained in the *STP.InBPDU* component, the severity of the condition, and the number of parameters in the event message are displayed. In this example, the severities of the events in the *STP.InBPDU* subcomponent range from error to debug-summary.

When you use the `details` keyword you will see the message text associated with the conditions. For example, if you want to see the message text and the parameters for the event condition *STP.InBPDU.Trace*, use the following command:

```
show log events stp.inbpdu.trace detail
```

The output produced by the command is similar to the following:

```
Comp    SubComp     Condition               Severity      Parameters
------- ----------- ----------------------- ------------- ----------
STP     InBPDU      Trace                   Debug-Verbose  2 total
                                                           0 - string
                                                           1 - string (printf)
                            Port=%0%: %1%
```

The `Comp` heading shows the component name, the `SubComp` heading shows the subcomponent (if any), the `Condition` heading shows the event condition, the `Severity` heading shows the severity assigned to this condition, the `Parameters` heading shows the parameters for the condition, and the text string shows the message that the condition will generate. The parameters in the text string (for example, `%0%` and `%1%` above) will be replaced by the values of these parameters when the condition is encountered, and output as the event message.

Filtering By Components and Conditions. You may want to send the messages that come from a specific component that makes up ExtremeWare XOS, or send the message generated by a specific condition. For example, you might want to send only the messages that come from the STP component, or send the message that occurs when the *IP.Forwarding.SlowPathDrop* condition occurs. Or you may want to exclude messages from a particular component or event. To do this, you will construct a filter that passes only the items of interest, and associate that filter with a target.

The first step is to create the filter using the `create log filter` command. You can create a filter from scratch, or copy another filter to use as a starting point. It may be easiest to copy an existing filter and modify it. To create a filter, use the following command:

```
create log filter <name> {copy <filter name>}
```

If you create a filter from scratch, it will initially block all events until you add events (either the events from a component or a specific event condition) to pass. You might create a filter from scratch if you wanted to pass a small set of events, and block most. If you want to exclude a small set of events, there is a default filter that passes events at or above the default severity threshold (unless the filter has been modified), named *DefaultFilter*, that you can copy to use as a starting point for your filter.

After you have created your filter, you can then configure filter items that include or exclude events from the filter. Included events are passed, excluded events are blocked. To configure your filter, use the following command:

```
configure log filter <name> [add | delete] {exclude} events [<event-condition> | [all
| <event-component>] {severity <severity> {only}}]
```

For example, if you create the filter *myFilter* from scratch, then issue the following command:

```
configure log filter myFilter add events stp
```

All STP component events will pass *myFilter* of at least the default threshold severity (for the STP component, the default severity threshold is `error`). You can further modify this filter by specifying additional conditions. For example, assume that *myFilter* is configured as before, and assume that you want to exclude the *STP.CreatPortMsgFail* event. To add that condition, use the following command:

```
configure log filter myFilter add exclude events stp.creatportmsgfail
```

You can also add events and subcomponents to the filter. For example, assume that *myFilter* is configured as before, and you want to include the STP.InBPDU subcomponent. To add that condition, use the following command:

```
configure log filter myFilter add events stp.inbpdu
```

You can continue to modify this filter by adding more filter items. The filters process events by comparing the event with the most recently configured filter item first. If the event matches this filter item, the incident is either included or excluded, depending on whether the `exclude` keyword was used. Subsequent filter items on the list are compared if necessary. If the list of filter items has been exhausted with no match, the event is excluded, and is blocked by the filter.

To examine the configuration of a filter, use the following command:

```
show log configuration filter {<filter name>}
```

The output produced by the command (for the earlier filter) is similar to the following:

```
Log Filter Name: myFilter
I/                                              Severity
E  Comp.   Sub-comp.   Condition                CEWNISVD
```

```
-  ------- ----------- ---------------------- --------
I  STP     InBPDU                            --------
E  STP                 CreatPortMsgFail      -E------
I  STP                                       --------

Include/Exclude: I - Include,  E - Exclude
Component Unreg: * - Component/Subcomponent is not currently registered
Severity Values: C - Critical,  E - Error,  W - Warning,  N - Notice,  I - Info
Debug Severity : S - Debug-Summary,  V - Debug-Verbose,  D - Debug-Data
                 + - Debug Severities, but log debug-mode not enabled
If Match parameters present:
Parameter Flags: S - Source,  D - Destination, (as applicable)
                 I - Ingress,  E - Egress,  B - BGP
Parameter Types: Port - Physical Port list,  Slot - Physical Slot #
                 MAC  - MAC address,  IP - IP Address/netmask,  Mask - Netmask
                 VID  - Virtual LAN ID (tag),  VLAN - Virtual LAN name
                 L4   - Layer-4 Port #,  Num  - Number,  Str  - String
                 Nbr  - Neighbor, Rtr  - Routerid, EAPS - EAPS Domain
                 Proc - Process Name
Strict Match   : Y - every match parameter entered must be present in the event
                 N - match parameters need not be present in the event
```

The show log configuration filter command shows each filter item, in the order that it will be applied and whether it will be included or excluded. The above output shows the three filter items, one excluding the event *STP.CreatPortMsgFail*, one including events from the *STP.InBPDU* component, and the next including the remaining events from the *STP* component. The severity value is shown as "*", indicating that the component's default severity threshold controls which messages are passed. The `Parameter(s)` heading is empty for this filter, since no match was configured for this filter. Matches are described in "Matching Expressions" next.

Each time a filter item is added to or deleted from a given filter, the events specified are compared against the current configuration of the filter to try to logically simplify the configuration. Existing items will be replaced by logically simpler items if the new item enables rewriting the filter. If the new item is already included or excluded from the currently configured filter, the new item is not added to the filter.

Matching Expressions

You can specify that messages that reach the target match a specified match expression. The message text is compared with the match expression to determine whether to pass the message on. To require that messages match a match expression, use the following command:

```
configure log target [console | memory-buffer | nvram | session | syslog [all |
<ipaddress> [local0 ... local7]]] match [any |<match-expression>]
```

The messages reaching the target will match the `match-expression`, a simple regular expression. The formatted text string that makes up the message is compared with the match expression, and is passed to the target if it matches. This command does not affect the filter in place for the target, so the match expression is only compared with the messages that have already passed the target's filter. For more information on controlling the format of the messages, see "Formatting Event Messages" on page 108.

Simple Regular Expressions. A simple regular expression is a string of single characters including the dot character (.), which are optionally combined with quantifiers and constraints. A dot matches any single character while other characters match only themselves (case is significant). Quantifiers include the star character (*) that matches zero or more occurrences of the immediately preceding token. Constraints include the caret character (^) that matches at the beginning of a message, and the currency character ($) that matches at the end of a message. Bracket expressions are not supported. There are a

number of sources available on the Internet and in various language references describing the operation of regular expressions. Table 20 shows some examples of regular expressions.

Table 20: Simple regular expressions

Regular Expression	Matches	Does Not Match
port	port 2:3 import cars portable structure	poor por pot
..ar	baar bazaar rebar	bar
port.*vlan	port 2:3 in vlan test add ports to vlan port/vlan	
myvlan$	delete myvlan error in myvlan	myvlan port 2:3 ports 2:4,3:4 myvlan link down

Matching Parameters

Rather than using a text match, EMS allows you to filter more efficiently based on the message parameter values. In addition to event components and conditions and severity levels, each filter item can also use parameter values to further limit which messages are passed or blocked. The process of creating, configuring, and using filters has already been described in “Filtering By Components and Conditions” on page 105, so this section describes matching parameters with a filter item. To configure a parameter match filter item, use the following command:

```
configure log filter <name> [add | delete] {exclude} events [<event-condition> | [all
| <event-component>] {severity <severity> {only}}] [match | strict-match] <type>
<value>
```

Each event in ExtremeWare XOS is defined with a message format and zero or more parameter types. The `show log events all` command can be used to display event definitions (the event text and parameter types). Only those parameter types that are applicable given the events and severity specified are exposed on the CLI. The syntax for the parameter types (represented by `<type>` in the command syntax above) is:

```
[bgp [neighbor | routerid] <ip address>
| {destination | source} [ipaddress <ip address> | L4-port <L4-port>| mac-address
<mac-address>]
| {egress | ingress} [slot <slot number> | ports <portlist>]
| netmask <netmask>
| number <number>
| string <match expression>
| vlan <vlan name>
| vlan tag <vlan tag>]
```

The `<value>` depends on the parameter type specified. As an example, an event may contain a physical port number, a source MAC address, and a destination MAC address. To allow only those radius incidents, of severity `notice` and above, with a specific source MAC address, use the following command:

```
configure log filter myFilter add events aaa.radius.requestInit secerity notice match
source mac-address 00:01:30:23:C1:00
```

The string type is used to match a specific string value of an event parameter, such as a user name. A string can be specified as a simple regular expression.

Match Versus Strict-Match. The `match` and `strict-match` keywords control the filter behavior for incidents whose event definition does not contain all the parameters specified in a `configure log filter events match` command. This is best explained with an example. Suppose an event in the *XYZ* component, named *XYZ.event5*, contains a physical port number, a source MAC address, but no destination MAC address. If you configure a filter to match a source MAC address and a destination MAC address, *XYZ.event5* will match the filter when the source MAC address matches regardless of the destination MAC address, since the event contains no destination MAC address. If you specify the `strict-match` keyword, then the filter will never match event *XYZ.event5*, since this event does not contain the destination MAC address.

In other words, if the `match` keyword is specified, an incident will pass a filter so long as all parameter values in the incident match those in the match criteria, but all parameter types in the match criteria need not be present in the event definition.

Formatting Event Messages

Event messages are made up of a number of items. The individual items can be formatted, however, EMS does not allow you to vary the order of the items. To format the messages for a particular target, use the following command:

```
configure log target [console | memory-buffer | nvram | session | syslog [all |
<ipaddress> [local0 ... local7]]]
format [timestamp [seconds | hundredths | none]
| date [dd-mm-yyyy | dd-Mmm-yyyy | mm-dd-yyyy | Mmm-dd | yyyy-mm-dd | none]
| severity
| event-name [component | condition | none | subcomponent]
| priority
| process-name
| process-slot
| source-line
```

Using the default format for the session target, an example log message might appear as:

```
05/29/2003 12:15:25.00 <Warn:SNTP.RslvSrvrFail> The SNTP server parameter value
(TheWrongServer.example.com) can not be resolved.
```

If you set the current session format using the following command:

```
configure log target session format timestamp seconds date mm-dd-yyyy event-name
component
```

The same example would appear as:

```
05/29/2003 12:16:36 <Warn:SNTP> The SNTP server parameter value
(TheWrongServer.example.com) can not be resolved.
```

To provide some detailed information to technical support, you set the current session format using the following command:

```
configure log target session format timestamp hundredths date mmm-dd event-name
condition source-line process-name
```

The same example would appear as:

```
May 29 12:17:20.11 SNTP: <Warn:SNTP.RslvSrvrFail> tSntpc: (sntpcLib.c:606) The SNTP
server parameter value (TheWrongServer.example.com) can not be resolved.
```

Displaying Real-Time Log Messages

You can configure the system to maintain a running real-time display of log messages on the console display or on a (telnet) session. To turn on the log display on the console, use the following command:

```
enable log target console
```

This setting may be saved to the FLASH configuration and will be restored on boot up (to the console-display session).

To turn on log display for the current session:

```
enable log target session
```

This setting only affects the current session, and is lost when you log off the session.

The messages that are displayed depend on the configuration and format of the target. For information on message filtering, see “Filtering Events Sent to Targets” on page 102. for information on message formatting, see “Formatting Event Messages” on page 108.

Displaying Events Logs

The log stored in the memory buffer and the NVRAM can be displayed on the current session (either the console display or telnet). To display the log, use the following command:

```
show log {messages [memory-buffer | nvram]} {events {<event-condition> |
<event-component>]} {<severity> {only}} {starting [date <date> time <time> | date
<date> | time <time>]} {ending [date <date> time <time> | date <date> | time <time>]}
{match <regex>} {chronological}
```

You can use many options to select the log entries of interest. You can select to display only those messages that conform to the specified:

- severity
- starting and ending date and time
- match expression

The displayed messages can be formatted differently from the format configured for the targets, and you can choose to display the messages in order of newest to oldest, or in chronological order (oldest to newest).

Uploading Events Logs

The log stored in the memory buffer and the NVRAM can be uploaded to a TFTP server. Use the following command to upload the log:

```
upload log <ipaddress> <filename> {messages [memory-buffer | nvram] {events
{<event-condition> | <event_component>}}} {<severity> {only}} {starting [date <date>
time <time> | date <date> | time <time>]} {ending [date <date> time <time> | date
<date> | time <time>]} {match <regex>} {chronological}
```

You must specify the TFTP host and the filename to use in uploading the log. There are many options you can use to select the log entries of interest. You can select to upload only those messages that conform to the specified:

- Severity
- Starting and ending date and time
- Match expression

The uploaded messages can be formatted differently from the format configured for the targets, and you can choose to upload the messages in order of newest to oldest, or in chronological order (oldest to newest).

Displaying Counts of Event Occurrences

EMS adds the ability to count the number of occurrences of events. Even when an event is filtered from all log targets, the event is counted. (The exception to this is events of any of the debug severities, which are only counted when the log debug mode is enabled.) To display the event counters, use the following command:

```
show log counters {<event condition> | [all | <event component>]} {include | notified
| occurred} {severity <severity> {only}}}
```

Two counters are displayed. One counter displays the number of times an event has occurred, and the other displays the number of times that notification for the event was made to the system for further processing. Both counters reflect totals accumulated since reboot or since the counters were cleared using the `clear log counters` or `clear counters` command.

This command also displays an included count (the column titled `In` in the output). The included count is the number of enabled targets receiving notifications of this event without regard to matching parameters.

The keywords `include`, `notified`, and `occurred` only display events with non-zero counter values for the corresponding counter.

Output of the command:

```
show log counters stp.inbpdu severity debug-summary
```

will be similar to the following:

```
Comp    SubComp     Condition               Severity      Occurred  In Notified
------- ----------- ----------------------- ------------ --------  -- --------
STP     InBPDU      Drop                    Error                0  Y         0
STP     InBPDU      Ign                     Debug-Summary        0  N         0
STP     InBPDU      Mismatch                Warning              0  Y         0

Occurred  : # of times this event has occurred since last clear or reboot
Flags     : (*) Not all applications responded in time with there count values
In(cluded): Set to Y(es) if one or more targets filter includes this event
Notified  : # of times this event has occurred when 'Included' was Y(es)
```

Output of the command:

```
show log counters stp.inbpdu.drop
```

will be similar to the following:

```
Comp    SubComp     Condition               Severity      Occurred  In Notified
------- ----------- ----------------------- ------------- --------  -- --------
STP     InBPDU      Drop                    Error                0  Y         0

Occurred  : # of times this event has occurred since last clear or reboot
Flags     : (*) Not all applications responded in time with there count values
In(cluded): Set to Y(es) if one or more targets filter includes this event
Notified  : # of times this event has occurred when 'Included' was Y(es)
```

Displaying Debug Information

By default, a switch will not generate events of severity `Debug-Summary`, `Debug-Verbose`, and `Debug-Data` unless the switch is in debug mode. Debug mode causes a performance penalty, so it should only be enabled for specific cases where it is needed. To place the switch in debug mode, use the following command:

```
enable log debug-mode
```

After debug mode has been enabled, any filters configured for your targets will still affect which messages are passed on or blocked.

9 Security

This chapter describes the following topics:

- Security Overview on page 113
- Network Access Security on page 113
  - IP Access Lists (ACLs) on page 113
- Switch Protection on page 119
  - Policies on page 120
- Management Access Security on page 129
  - Authenticating Users Using RADIUS or TACACS+ on page 129

Security Overview

Extreme Networks products incorporate a number of features designed to enhance the security of your network. No one feature can insure security, but by using a number of features in concert, you can substantially improve the security of your network. The features described in this chapter are part of an overall approach to network security

Network Access Security

Network access security features control devices accessing your network. In this category are the following features:

- IP Access Lists (ACLs)

IP Access Lists (ACLs)

IP access lists consist of IP access rules and are used to perform packet filtering and forwarding decisions on incoming traffic. Each packet arriving on an ingress port is compared to the access list applied to that port and is either permitted or denied. Permitted and denied (dropped) packets can also be counted. Using access lists has no impact on switch performance.

Access lists are typically applied to traffic that crosses layer 3 router boundaries, but it is possible to use access lists within a layer 2 VLAN.

Access lists in ExtremeWare XOS apply to all traffic. This is somewhat different from the behavior in ExtremeWare. For example, if you deny all the traffic to a port, no traffic, including control packets, such as OSPF or RIP, will reach the switch and the adjacency will be dropped. You must explicitly allow those type of packets (if desired). In ExtremeWare, an access list that denied “all” traffic would allow control packets (those bound for the CPU) to reach the switch.

Access lists are often referred to as Access Control Lists (ACLs).

The following sections apply to IP access lists:

- Creating IP Access Lists on page 114
- ACL File Syntax on page 114
- Example ACL Rule Entries on page 117
- Using Access Lists on the Switch on page 118
- Displaying and Clearing ACL Counters on page 119

Creating IP Access Lists

ACLs are created by writing a text file containing a number of rule entries. Name the text file with the ACL name and use “.pol” as the filename extension. For example, the ACL name “zone3” refers to the text file “zone3.pol”. Any common text editor can be used to create an access list file. The file is then transferred to the switch using TFTP, and applied to some or all ports on the switch.

ACL File Syntax

The ACL file contains one or more rule entries. Each rule entry consists of:

- a rule entry name, unique within the same ACL.
- zero or more match conditions. If no match condition is specified, all packets are matched.
- zero or one action. If no action is specified, the packet is permitted by default.
- zero or more action modifiers.

Each rule entry in the file uses the following syntax:

```
entry <entry-name>{
   if  {
      <match-conditions>;
   } then {
      <action>;
      <action-modifiers>;
   }
}
```

Here is an example of a rule entry:

```
entry  udpacl {
   if  {
      source-address 10.203.134.0/24;
      destination-address 140.158.18.16/32;
```

```
      protocol  udp;
      source-port 190;
      destination-port  1200-1400;
   } then {
      permit;
   }
}
```

ACL rule entries are evaluated in order, from the beginning of the file to the end, as follows:

- If the packet matches all the match conditions, the action in the then statement is taken and evaluation process terminates.
- If a rule entry does not contain any match condition, the packet is considered to match and the action in the rule entry's then statement is taken and evaluation process terminates.
- If the packet matches all the match conditions, and if there is no action specified in the then statement, the action permit is taken by default.
- If the packet does not match all the match conditions, the next rule entry in the ACL is evaluated.
- This process continues until either the packet matches all the match conditions in one of the subsequent rule entries or there are no more entries.
- If a packet passes through all the rule entries in the ACL without matching any of them, it is permitted.

Often an ACL will have a rule entry at the end of the ACL with no match conditions. This entry will match any packets not otherwise processed, so that user can specify an action to overwrite the default permit action.

Match Conditions. Multiple, single, or zero match conditions can be specified. If no match condition is specified, all packets match the rule entry. Among the match conditions commonly used are:

- IP source address and mask
- IP destination address and mask
- TCP or UDP source port range
- TCP or UDP destination port range

Table 21describes all the possible match conditions.

Actions. The action is either `permit`, `deny`, or no action specified. No action specified permits the packet. The deny action drops the packet.

Action Modifier. The action modifier is `count`. The count action increments the counter named in the action modifier.

Table 21 lists the match conditions that can be used with ACLs. The conditions are case-insensitive; for example, the match condition listed in the table as `TCP-flags` can also be written as `tcp-flags`. Within Table 21 are five different data types used in matching packets. Table 22 lists the data types, and details on using them.

Table 21: ACL Match Conditions

Match Conditions	Description	Applicable IP Protocols
source-address <prefix>	IP source address and mask.	All IP
destination-address <prefix>	IP destination address and mask.	All IP
protocol <number>	IP protocol field. In place of the numeric value, you can specify one of the following text synonyms (the field values are also listed): egp(8), esp(5), gre(47), icmp(1), igmp(2),ipip(4), ipv6(41), ospf(89),pim(102), rsvp(46),tcp(6), or udp(17)	All IP
Source-port {<number> | <range>}	TCP or UDP source port. In place of the numeric value, you can specify one of the text synonyms listed under destination port.	
Destination-port {<number> | <range>}	TCP or UDP destination port. Normally, you specify this match in conjunction with the protocol match to determine which protocol is being used on the port.In place of the numeric value, you can specify one of the following text synonyms (the field values are also listed): afs(1483), bgp(179), biff(512), bootpc(68), bootps(67), cmd(514), cvspserver(2401), DHCP(67), domain(53), eklogin(2105), ekshell(2106), exec(512), finger(79), ftp(21), ftp-data(20), http(80), https(443), ident(113), imap(143), kerberos-sec(88), klogin(543), kpasswd(761), krb-prop(754), krbupdate(760), kshell(544), idap(389), login(513), mobileip-agent(434), mobileip-mn(435), msdp(639), netbios-dgm(138), netbios-ns(137), netbios-ssn(139), nfsd(2049), nntp(119), ntalk(518), ntp(123), pop3(110), pptp(1723), printer (515), radacct(1813), radius(1812), rip(520), rkinit(2108), smtp(25), snmp(161), snmptrap(162), snpp(444), socks(1080), ssh(22), sunrpc(111), syslog(514), tacacs-ds(65), talk(517), telnet(23), tftp(69), timed(525), who(513), xdmcp(177), zephyr-clt(2103), or zephyr-hm(2104).	
TCP-flags <bitfield>	TCP flags. Normally, you specify this match in conjunction with the protocol match statement. In place of the numeric value, you can specify one of the following text synonyms (the field values are also listed): ACK(0x10), FIN(0x01), PUSH(0x08), RST(0x04), SYN(0x02), URG(0x20), SYN_ACK(0x12).	TCP
IGMP-msg-type <number>	IGMP message type. Possible values and text synonyms: v1-report(0x12), v2-report(0x16), v3-report(0x22), V2-leave (0x17)query(0x11)	IGMP
ICMP-type <number>	ICMP type field. Normally, you specify this match in conjunction with the protocol match statement. In place of the numeric value, you can specify one of the following text synonyms (the field values are also listed): echo-reply(0), echo-request(8), info-reply(16), info-request(15), mask-request(17), mask-reply(18), parameter-problem(12), redirect(5), router-advertisement(9), router-solicit(10), source-quench(4), time-exceeded(11), timestamp(13), timestamp-reply(14), or unreachable(3).	ICMP

Table 21: ACL Match Conditions (continued)

Match Conditions	Description	Applicable IP Protocols
ICMP-code <number>	ICMP code field. This value or keyword provides more specific information than the icmp-type. Since the value's meaning depends upon the associated icmp-type, you must specify the icmp-type along with the icmp-code.In place of the numeric value, you can specify one of the following text synonyms (the field values also listed). The keywords are grouped by the ICMP type with which they are associated: Parameter-problem: ip-header-bad(0), required-option-missing(1) Redirect: redirect-for-host (1), redirect-for-network (2), redirect-for-tos-and-host (3), redirect-for-tos-and-net (2) Time-exceeded: ttl-eq-zero-during-reassembly(1), ttl-eq-zero-during-transit(0) Unreachable: communication-prohibited-by-filtering(13), destination-host-prohibited(10), destination-host-unknown(7), destination-network-prohibited(9), destination-network-unknown(6), fragmentation-needed(4), host-precedence-violation(14), host-unreachable(1), host-unreachable-for-TOS(12), network-unreachable(0), network-unreachable-for-TOS(11), port-unreachable(3), precedence-cutoff-in-effect(15), protocol-unreachable(2), source-host-isolated(8), source-route-failed(5)	ICMP

Along with the data types described in Table 22, you can use the operators <, <=, >, and >= to specify match conditions. For example, the match condition, `source-port >190`, will match packets with a source port greater than 190.

Table 22: ACL Match Condition Data Types

Condition Data Type	Description
prefix	IP source and destination address prefixes. To specify the address prefix, use the notation `prefix/prefix-length`. For a host address, `prefix-length` should be set to 32.
number	Numeric value. This can be TCP or UDP source and destination port number, IP protocol number, etc.
range	A range of numeric values. To specify the numeric range, use the notation: `number - number`
bit-field	Used to match specific bits in an IP packet, such as TCP flags and the fragment flag.
mac-address	6-byte hardware address.

Example ACL Rule Entries

The following entry accepts all the UDP packets from the 10.203.134.0/24 subnet that are destined for the host 140.158.18.16, with source port 190 and a destination port in the range of 1200 - 1400:

```
entry  udpacl {
   if  {
```

```
      source-address 10.203.134.0/24;
      destination-address 140.158.18.16/32;
      protocol  udp;
      source-port 190;
      destination-port  1200-1400;
   } then {
      accept;
   }
}
```

The following rule entry accepts TCP packets from the 10.203.134.0/24 subnet with a source port larger than 190 and ACK & SYN bits set, and also increments the counter *tcpcnt*:

```
entry  tcpacl {
   if  {
      source-address 10.203.134.0/24;
      protocol  TCP;
      source-port  >190;
      tcp-flags  syn_ack;
   } then {
      accept;
      count tcpcnt ;
   }
}
```

The following example denies ICMP echo request packets from the 10.203.134.0/24 subnet, and increments the counter *icmpcnt*:

```
entry  icmp {
   if  {
      source-address 10.203.134.0/24;
      protocol  icmp;
      icmp-type  echo-request;
   } then {
      deny;
      count icmpcnt;
   }
}
```

The following entry denies every packet and increments the counter *default*:

```
entry  default {
   if  {

   } then {
      deny;
      count default;
   }
}
```

Using Access Lists on the Switch

After the access list file has been transferred to the switch, it can be checked to see if it is syntactically correct. Since an ACL is a type of policy, use the following command to check the ACL syntax:

```
check policy <policy-name>
```

After the ACL has been checked, it can be applied to an interface. To apply an ACL, use the following command:

```
configure access-list <aclname> [any | ports <portlist>] {ingress}
```

If you use the any keyword, the ACL is applied to all the interfaces, and is referred to as the wildcard ACL. This ACL is evaluated for ports without a specific ACL applied to it, and is also applied to packets that do not match the ACL applied to the interface.

If an ACL is already configured on an interface, the command will be rejected and an error message displayed.

To remove an ACL from an interface, use the following command:

```
unconfigure access-list {any | ports <portlist>} {ingress}
```

To display which interfaces have ACLs configured, and which ACL is on which interface, use the following command:

```
show access-list {<aclname>}
```

Displaying and Clearing ACL Counters

To display the access list counters, use the following command:

```
show access-list counter {<countername>} [any | ports <portlist>] {ingress}
```

To clear the access list counters, use the following command:

```
clear access-list counter {<countername>} [any | ports <portlist>] {ingress}
```

Switch Protection

Switch protection features enhance the robustness of switch performance. In this category are the following features:

- Routing Access Profiles
- Route Maps
- Policies

In ExtremeWare XOS, all of these features are supported by the concept of a policy. If you have used any of these features in the past, you will now use policies to achieve the same results.

Routing access profiles are used to control the advertisement or recognition of routing protocols, such as RIP, OSPF, IS-IS, or BGP. Routing access profiles can be used to ‘hide’ entire networks, or to trust only specific sources for routes or ranges of routes. The capabilities of routing access profiles are specific to the type of routing protocol involved, but are sometimes more efficient and easier to implement than access lists.

Route maps are used to modify or filter routes. They are also used to modify or filter routing information.

Policies

Policies are a more general concept than routing access profiles and route maps. ExtremeWare XOS uses policies to implement routing access profiles and route maps. A central manager processes policies, and various policy clients, such as BGP or OSPF, get the policies from the central manager.

The following sections apply to creating and using policies:

- Creating Policies on page 120
- Policy File Syntax on page 120
- Policy Examples on page 125
- Using Policies on page 129

Creating Policies

Policies are created by writing a text file containing a number of rule entries. Name the text file with the policy name and use ".pol" as the filename extension. For example, the policy name "boundary" refers to the text file "boundary.pol". Any common text editor can be used to create a policy file. The file is then transferred to the switch using TFTP, and then applied.

To transfer policy files to the switch, use the following command:

```
tftp [<ip_address> | <host_name>] {-v <vr_id>} [-g | -p] [{-l <local_file>} {-r
<remote_file>} | {-r <remote_file>} {-l <local_file>}]
```

Policy File Syntax

The policy file contains one or more policy entries. Each policy entry consists of:

- a policy entry name, unique within the same policy.
- zero or one match type. If no type is specified, the match type is all, so all match conditions must be satisfied.
- zero or more match conditions. If no match condition is specified, all are matched.
- zero or more actions. If no action is specified, no action is taken, and processing continues.

Each policy entry in the file uses the following syntax:

```
entry <entry-name>{
   if <match-type> {
      <match-conditions>;
   } then {
      <action>;
   }
}
```

Here is an example of a policy entry:

```
entry ip_entry {
   if match any {
      nlri  10.203.134.0/24;
      nlri  10.204.134.0/24;
   } then {
      next-hop   192.168.174.92;
```

```
    origin   egp;
  }
}
```

Policy entries are evaluated in order, from the beginning of the file to the end, as follows:

- If a match occurs, the action in the then statement is taken:
  - if the action contains an explicit permit or deny, the evaluation process terminates.
  - if the action does not contain an explicit permit or deny, then the action is an implicit permit, and the evaluation process terminates.
- If a match does not occur, then the next policy entry is evaluated.
- If no match has occurred after evaluating all policy entries, the default action is deny.

Often a policy will have a rule entry at the end of the policy with no match conditions. This entry will match anything not otherwise processed, so that user can specify an action to override the default deny action.

The next sections list detailed information about policy match conditions, about matching BGP AS paths, and about action statements. For information on those subjects, see the following sections:

- “Policy Match Conditions” on page 121
- “Autonomous System Expressions” on page 122
- “Policy Action Statements” on page 124

Policy Match Conditions

Table 23 lists the possible policy entry match conditions.

Table 23: Policy Match Conditions

Match Condition	Description
as-path [<as-number> | <as-path-regular-expression>];	Where <as-number> is a valid Autonomous system number in the range [1 - 65535]. <as-path-regular-expression> is a multi-character regular expression (with 2-byte unsigned Integer being an Atom). Regular expression will consist of the AS-Numbers and various regular expression symbols. Regular expressions must be enclosed in double quotes ("").
community [no-advertise | no-export | no-export-subconfed | number <community_num> | <community_regular_expression> | <as_num> : <num>];	"no-advertise", "no-export" and "no-export-subconfed" are the standard communities defined by RFC. <community_num> is a four byte unsigned integer, <as_num> is a two byte AS-Number and <num> is the 2-bytes community number. Community regular expression is a multi-character regular expression (with four byte unsigned integer being an Atom). Regular expression is enclosed in double quotes ("").
med <number>;	<number> is a four byte unsigned integer.
next-hop [<ipaddress> {<ipaddress1> <ipaddress2> <ipaddress3> ..} | <ipaddress-regular-expression>];	<ipaddress> is a valid IP address in dotted decimal format. User can supply multiple IP addresses (separated by space) to match against the next hop.

Table 23: Policy Match Conditions (continued)

Match Condition	Description
nlri [<ipaddress> | any]/<mask-length> {exact}; nlri [<ipaddress> | any] mask <mask> {exact};	Where, <ipaddress> and <mask> are in dotted decimal format, <mask-length> is an integer in the range [0 - 32]. Keyword any matches any IP address with a given (or larger) mask/mask-length.
origin [igp | egp | incomplete];	IGP, EGP and incomplete are the BGP route origin values.
tag <number>;	<number> is a four byte unsigned number.
route-origin [direct | static | icmp | egp | ggp | hello | rip | isis | esis | cisco-igrp | ospf | bgp | idrp | dvmrp | mospf | pim-dm | pim-sm | ospf-intra | ospf-inter | ospf-as-external | ospf-extern-1 | ospf-extern-2 | bootp | e-bgp | i-bgp | mbgp | i-mbgp | e-mbgp | isis-level-1 | isis-level-2 | isis-level-1-external | isis-level-2-external]	Matches the origin (different from BGP route origin) of a route. A match statement "route-origin bgp" will match routes whose origin are "I-bgp" or "e-bgp" or "I-mbgp" or "e-mbgp". Similarly, the match statement "route-origin ospf" will match routes whose origin is "ospf-inta" or "ospf-inter" or "ospf-as-external" or "ospf-extern-1" or "ospf-extern-2"

Autonomous System Expressions

The AS-path keyword uses a regular expression string to match against the AS path. Table 24 lists the regular expressions that can be used in the match conditions for BGP AS path and community. Table 25 list examples of regular expressions and the AS paths they match.

Table 24: Autonomous System Regular Expression Notation

Character	Definition
N	As number
N_1 - N_2	Range of AS numbers, where N_1 and N_2 are AS numbers and $N_1 < N_2$
[N_x ... N_y]	Group of AS numbers, where N_x and N_y are AS numbers or a range of AS numbers
[^N_x ... N_y]	Any AS numbers other than the ones in the group
.	Matches any number
^	Matches the beginning of the AS path
$	Matches the end of the AS path
–	Matches the beginning or end, or a space
-	Separates the beginning and end of a range of numbers
*	Matches 0 or more instances
+	Matches 1 or more instances
?	Matches 0 or 1 instance
{	Start of AS SET segment in the AS path
}	End of AS SET segment in the AS path
(	Start of a confederation segment in the AS path
)	End of a confederation segment in the AS path

Table 25: Policy Regular Expression Examples

Attribute	Regular Expression	Example Matches
AS path is 1234	“1234”	1234

Table 25: Policy Regular Expression Examples

Attribute	Regular Expression	Example Matches
Zero or more occurrences of AS number 1234	“1234*”	1234 1234 1234
Start of As path set	“10 12 { 34”	10 12 34 { 99 33 10 12 { 34 37
End of As path set	“12 } 34”	12 } 34 56
Path that starts with 99 followed by 34	“^99 34 “	99 34 45
Path that ends with 99	“99 $”	45 66 99
Path of any length that begins with AS numbers 4, 5, 6	“4 5 6 .*”	4 5 6 4 5 6 7 8 9
Path of any length that ends with AS numbers 4, 5, 6	“.* 4 5 6”	4 5 6 1 2 3 4 5 6

Here are some additional examples of using regular expressions in the AS-Path statement.

The following AS-Path statement matches AS paths that contain only (begin and end with) AS number 65535:

```
as-path "^65535$"
```

The following AS-Path statement matches AS paths beginning with AS number 65535, ending with AS number 14490, and containing no other AS paths:

```
as-path "^65535 14490$"
```

The following AS-Path statement matches AS paths beginning with AS number 1, followed by any AS number from 2 - 8, and ending with either AS number 11, 13, or 15:

```
as-path "^1 2-8 [11 13 15]$"
```

The following AS-Path statement matches AS paths beginning with AS number 111 and ending with any AS number from 2 - 8:

```
as-path "111 [2-8]"
```

The following AS-Path statement matches AS paths beginning with AS number 111 and ending with any additional AS number, or beginning and ending with AS number 111:

```
as-path "111 .?"
```

Policy Action Statements

Table 26 lists the possible action statements. These are the actions taken when the policy match conditions are met in a policy entry.

Table 26: Policy Actions

Action	Description
accounting-index <number> value <number>;	Sets the accounting index for a route. This is used in the import policy.
as-path "<as_num> {<as_num1> <as_num2> <as_num3> .... <as_numN>}";	Prepends the entire list of as-numbers to the as-path of the route.
community [no-advertise | no-export | no-export-subconfed | <community_num> {<community_num1> <community_num2> .... <community_numN>} | <as_num> : <community_num> [<as_num1> <community_num1> <as_num2> <community_num2> ....}];	Replaces the existing community attribute of a route by the communities specified by the action statement. Communities must be enclosed in double quotes ("").
community [add | delete] [no-advertise | no-export | no-export-subconfed | <community_num> {<community_num1> <community_num2> .... <community_numN>} | <as_num> : <community_num> {<as_num1> <community_num1> <as_num2> <community_num2> ....}];	Adds/deletes communities to/from a route's community attribute. Communities must be enclosed in double quotes ("").
community remove;	Strips off the entire community attribute from a route. Communities must be enclosed in double quotes ("").
cost <cost(0-4261412864)>;	Sets the cost/metric for a route.
cost-type {ase-type-1 | ase-type-2 | external | internal};	Sets the cost type for a route.
dampening half-life <minutes (1-45)> reuse-limit <number (1-20000)> suppress-limit <number (1-20000)> max-suppress <minutes (1-255)>;	Sets the BGP route flap dampening parameters.
deny;	Deny the route.
local-preference <number>;	Sets the BGP local preference for a route.
med {add | delete} <number>;	Performs MED arithmetic. "Add" means the value of the MED in the route will be incremented by <number> and "delete" means the value of the MED in the route will be decremented by <number>.
med {internal | remove};	"Internal" means that IGP distance to the next hop will be taken as the MED for a route. "Remove" means take out the MED attribute from the route.
med set <number>;	Sets the MED attribute for a route.
next-hop <ipaddress>;	Sets the next hop attribute for a route.
nlri [<ipaddress> | any]/<mask-length> {exact}; nlri [<ipaddress> | any] mask <mask> {exact};	These set statements are used for building a list of IP addresses. This is used by PIM to set up the RP list.
origin {igp | egp | incomplete};	Sets the BGP route origin values.
permit;	Permit the route.
tag <number>;	Sets the tag number for a route.
weight <number>	Sets the weight for a route.

Policy Examples

The following sections contain examples of policies. The examples are:

- “Translating an Access Profile to a Policy” on page 125
- “Translating a Route Map to a Policy” on page 127

Translating an Access Profile to a Policy

You may be more familiar with using access profiles on other Extreme Networks’ switches. This example shows the policy equivalent to an access-profile.

ExtremeWare Access-Profile:

```
Seq_No Action IP Address       IP Mask         Exact
5      permit 22.16.0.0        255.252.0.0        No
10     permit 192.168.0.0      255.255.192.0      Yes
15     deny   any              255.0.0.0          No
20     permit 10.10.0.0        255.255.192.0      No
25     deny   22.44.66.0       255.255.254.0      Yes
```

Equivalent ExtremeWare XOS Policy-Map definition:

```
entry  entry-5  {
   If  {
      nlri    22.16.0.0/14;
   }
   then  {
      permit;
   }
}

entry  entry-10  {
   if  {
      nlri   192.168.0.0/18 exact;
   }
   then  {
      permit;
   }
}

entry  entry-15  {
   if  {
      nlri   any/8;
   }
   then  {
      deny;
   }
}

entry  entry-20  {
   if  {
      nlri   10.10.0.0/18;
}
```

```
then  {
   permit;
}
}

entry  entry-25  {
   if  {
      nlri   22.44.66.0/23  exact;
   }
   then  {
      deny;
   }
}
```

The policy above can be optimized by combining some of the if into a single expression. The compact form of the policy will look like this:

```
entry  permit_entry  {
   If match any {
      nlri    22.16.0.0/14;
      nlri   192.168.0.0/18 exact ;
      nlri   10.10.0.0/18;
   }
   then  {
      permit;
   }
}

entry  deny_entry  {
   if match any {
      nlri   any/8;
      nlri   22.44.66.0/23  exact;
   }
   then  {
      deny;
   }
}
```

Translating a Route Map to a Policy

You may be more familiar with using route maps on other Extreme Networks' switches. This example shows the policy equivalent to a route map.

ExtremeWare Route Map:

```
Route Map : rt
    Entry : 10     Action : permit
        match origin incomplete
    Entry : 20     Action : deny
        match community  6553800
    Entry : 30     Action : permit
        match med  30
        set next-hop 10.201.23.10
        set as-path 20
        set as-path 30
        set as-path 40
        set as-path 40
    Entry : 40     Action : permit
        set local-preference 120
        set weight 2
    Entry : 50     Action : permit
        match origin incomplete
        match community  19661200
        set dampening half-life 20 reuse-limit 1000 suppress-limit 3000 max-suppress
40
    Entry : 60     Action : permit
        match next-hop 192.168.1.5
        set community add 949616660
```

Here is the equivalent policy:

```
entry  entry-10  {
   If  {
      origin    incomplete;
   }
   then  {
      permit;
   }
}

entry  entry-20  {
   if  {
      community   6553800;
   }
   then  {
      deny;
   }
}

entry  entry-30  {
   if  {
      med   30;
   }
   then  {
      next-hop 10.201.23.10;
```

```
      as-path 20;
      as-path 30;
      as-path 40;
      as-path 40;
      permit;
   }
}

entry  entry-40  {
   if  {

   }
   then  {
      local-preference 120;
      weight 2;
      permit;
}
}

entry  entry-50 match any {
   if  {
      origin incomplete;
      community 19661200;
   }
   then  {
      dampening half-life 20 reuse-limit 1000 suppress-limit 3000 max-suppress 40
      permit;
   }
}

entry  entry-60 {
   if  {
      next-hop 192.168.1.5;
   }
   then  {
      community  add  949616660;
      permit;
   }
}

entry  deny_rest {
   if  {

   }
   then  {
      deny;
   }
}
```

Using Policies

After the policy file has been transferred on the switch, it can be checked to see if it is syntactically correct. Use the following command to check the policy syntax:

```
check policy <policy-name>
```

To apply a policy, use the command appropriate to the client. Some examples include:

```
configure bgp import-policy [<policy-name> | none]
configure bgp neighbor [<remoteaddr> | all] {address-family [ipv4-unicast |
ipv4-multicast]} route-policy [in | out] [none | <policy>]
configure bgp peer-group <peer-group-name> route-policy [in | out] [none | <policy>]
configure ospf area <area-identifier> external-filter [<policy-map> |none]
configure ospf area <area-identifier> interarea-filter [<policy-map> | none]
configure rip import-policy [<policy-name> | none]
configure rip vlan [<vlan-name> | all] route policy [in | out] [<policy-name> | none]
configure rip [vlan <vlan-name> | all] trusted-gateway [<policy-name> | none]
```

To remove a policy, use the none option.

Refreshing Policies

When a policy file is changed (adding, deleting an entry, adding / deleting / modifying a statement etc.), the new file can be downloaded to the switch and the user must refresh the policy so that the latest copy of policy will be used.

When the policy is refreshed, the policy file is read, processed, and stored in the server database. Any clients that use the policy will also be updated. Use the following command to refresh the policy:

```
refresh policy <policy-name>
```

Saving Policies

A policy file in the server database can be saved back to a text file. You can save the policy to the original file name, or save it to a different file name. Use the following command to save the policy:

```
save policy <policy-name> {<newName>}
```

Management Access Security

Management access security features control access to the management functions available on the switch. These features help insure that any configuration changes to the switch can only be done by authorized users. In this category are the following features:

- Authenticating Users Using RADIUS or TACACS+

Authenticating Users Using RADIUS or TACACS+

ExtremeWare XOS provides two methods to authenticate users who login to the switch:

- RADIUS
- TACACS+

RADIUS

Remote Authentication Dial In User Service (RADIUS, RFC 2138) is a mechanism for authenticating and centrally administrating access to network nodes. The ExtremeWare XOS RADIUS implementation allows authentication for Telnet or console access to the switch.

NOTE

You cannot configure RADIUS and TACACS+ at the same time.

You can define a primary and secondary RADIUS server for the switch to contact. When a user attempts to login using Telnet, http, or the console, the request is relayed to the primary RADIUS server, and then to the secondary RADIUS server, if the primary does not respond. If the RADIUS client is enabled, but access to the RADIUS primary and secondary server fails, the switch uses its local database for authentication.

The privileges assigned to the user (admin versus nonadmin) at the RADIUS server take precedence over the configuration in the local switch database.

To configure the RADIUS servers, use the following command:

```
configure radius [primary | secondary] server [<ipaddress> | <hostname>] {<udp_port>}
client-ip [<ipaddress>] {vr <vr_name>}
```

To configure the timeout if a server fails to respond, use the following command:

```
configure radius timeout <seconds>
```

Configuring the Shared Secret Password

In addition to specifying the RADIUS server IP information, RADIUS also contains a means to verify communication between network devices and the server. The *shared secret* is a password configured on the network device and RADIUS server, used by each to verify communication.

To configure the shared secret for RADIUS servers, use the following command:

```
configure radius [primary | secondary] shared-secret [<string>]
```

Enabling and Disabling RADIUS

After server information is entered, you can start and stop RADIUS authentication as many times as necessary without needing to reconfigure server information.

To enable RADIUS authentication, use the following command:

```
enable radius
```

To disable RADIUS authentication, use the following command:

```
disable radius
```

Configuring RADIUS Accounting

Extreme switches are capable of sending RADIUS accounting information. As with RADIUS authentication, you can specify two servers for receipt of accounting information.

To specify RADIUS accounting servers, use the following command:

```
configure radius-accounting [primary | secondary] server [<ipaddress> | <hostname>]
{<tcp_port>} client-ip [<ipaddress>] {vr <vr_name>}
```

To configure the timeout if a server fails to respond, use the following command:

```
configure radius-accounting timeout <seconds>
```

RADIUS accounting also makes use of the shared secret password mechanism to validate communication between network access devices and RADIUS accounting servers.

To specify shared secret passwords for RADIUS accounting servers, use the following command:

```
configure radius-accounting [primary | secondary] shared-secret [<string>]
```

After you configure RADIUS accounting server information, you must enable accounting before the switch begins transmitting the information. You must enable RADIUS authentication for accounting information to be generated. You can enable and disable accounting without affecting the current state of RADIUS authentication.

To enable RADIUS accounting, use the following command:

```
enable radius-accounting
```

To disable RADIUS accounting, use the following command:

```
disable radius-accounting
```

Configuring RADIUS

You can define primary and secondary server communication information, and for each RADIUS server, the RADIUS port number to use when talking to the RADIUS server. The default port value is 1812for authentication and 1813 for accounting. The client IP address is the IP address used by the RADIUS server for communicating back to the switch.

RADIUS RFC 2138 Attributes

The RADIUS RFC 2138 optional attributes supported are as follows:

- User-Name
- User-Password
- Service-Type
- Login-IP-Host

Using RADIUS Servers with Extreme Switches

Extreme Networks switches have two levels of user privilege:

- Read-only
- Read-write

Because no CLI commands are available to modify the privilege level, access rights are determined when you log in. For a RADIUS server to identify the administrative privileges of a user, Extreme switches expect a RADIUS server to transmit the Service-Type attribute in the Access-Accept packet, after successfully authenticating the user.

Extreme switches grant a RADIUS-authenticated user read-write privilege if a Service-Type value of 6 is transmitted as part of the Access-Accept message from the Radius server. Other Service-Type values, or no value, result in the switch granting read-only access to the user. Different implementations of RADIUS handle attribute transmission differently. You should consult the documentation for your specific implementation of RADIUS when you configure users for read-write access.

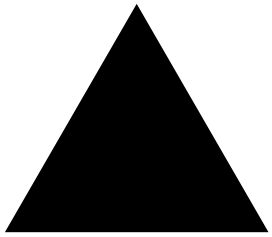
Configuring TACACS+

Terminal Access Controller Access Control System Plus (TACACS+) is a mechanism for providing authentication, authorization, and accounting on a centralized server, similar in function to RADIUS. The ExtremeWare XOS version of TACACS+ is used to authenticate prospective users who are attempting to administer the switch. TACACS+ is used to communicate between the switch and an authentication database.

NOTE

You cannot use RADIUS and TACACS+ at the same time.

You can configure two TACACS+ servers, specifying the primary server address, secondary server address, and TCP port number to be used for TACACS+ sessions.



Part 2

Using Switching and Routing Protocols

10 Spanning Tree Protocol (STP)

This chapter covers the following topics:

- Overview of the Spanning Tree Protocol on page 135
- Spanning Tree Domains on page 137
- STP Configurations on page 142
- Per-VLAN Spanning Tree on page 148
- Rapid Spanning Tree Protocol on page 148
- STP Rules and Restrictions on page 159
- Configuring STP on the Switch on page 159
- Displaying STP Settings on page 163

Using the Spanning Tree Protocol (STP) functionality of the switch makes your network more fault tolerant. The following sections explain more about STP and the STP features supported by ExtremeWare XOS.

NOTE

STP is a part of the 802.1d bridge specification defined by the IEEE Computer Society. To explain STP in terms used by the 802.1d specification, the switch will be referred to as a bridge.

Overview of the Spanning Tree Protocol

STP is a bridge-based mechanism for providing fault tolerance on networks. STP allows you to implement parallel paths for network traffic, and ensure that redundant paths are:

- Disabled when the main paths are operational.
- Enabled if the main path fails.

STP Terms

Table 27 describes the terms associated with STP.

Table 27: STP terms

Term	Description
autobind	If enabled, autobind automatically adds or removes ports from the STPD. If ports are added to the carrier VLAN, the member ports of the VLAN are automatically added to the STPD. If ports are removed from the carrier VLAN, those ports are also removed from the STPD. For more information about the autobind feature and adding ports to an STPD, see “Binding Ports” on page 140.
carrier VLAN	Carrier VLANs define the scope of the Spanning Tree Domain (STPD) including the physical and logical ports that belong to the STPD and the 802.1q tag used to transport EMISTP or PVST+ encapsulated BPDUs. Only one carrier VLAN can exist in any given STP domain. For more information about carrier VLANs, see “Member VLANs” on page 137.
default encapsulation mode	Default encapsulation allows you to specify the type of BPDU encapsulation to use for all ports added to a given STPD, not just one individual port. The three encapsulation modes are: • 802.1d—This mode is used for backward compatibility with previous STP versions and for compatibility with third-party switches using IEEE standard 802.1d. • EMISTP—Extreme Multiple Instance Spanning Tree Protocol (EMISTP) mode is an extension of STP that allows a physical port to belong to multiple STPDs by assigning the port to multiple VLANs. • PVST+ —This mode implements PVST+ in compatibility with third-party switches running this version of STP. For more information about how to configure the default encapsulation mode, see “Encapsulation Modes” on page 139.
encapsulation mode	You can configure ports within an STPD to accept specific BPDU encapsulations. The three encapsulation modes are: • 802.1d—This mode is used for backward compatibility with previous STP versions and for compatibility with third-party switches using IEEE standard 802.1d. • EMISTP—Extreme Multiple Instance Spanning Tree Protocol (EMISTP) mode is an extension of STP that allows a physical port to belong to multiple STPDs by assigning the port to multiple VLANs. • PVST+ —This mode implements PVST+ in compatibility with third-party switches running this version of STP. For more information about how to configure encapsulation modes, see “Encapsulation Modes” on page 139.
protected VLAN	Protected VLANs are the other VLANs that are members of the STPD but do not define the scope of the STPD. Protected VLANs do not transmit or receive STP BPDUs, but they are affected by STP state changes and inherit the state of the carrier VLAN. Also known as non-carrier VLANs. For more information about protected VLANs, see “Member VLANs” on page 137.
Spanning Tree Domain	An STP instance that contains one or more VLANs. The switch can run multiple Spanning Tree Domains (STPDs). For more information about STPD, see “Spanning Tree Domains” on page 137.
STPD mode	The mode of operation for the STPD. The two modes of operation are: • 802.1d—Compatible with legacy STP and other devices using the IEEE 802.1d standard) • 802.1w—Compatible with Rapid Spanning Tree (RSTP) For more information about how to configure STPD modes, see “STPD Modes” on page 138.

Spanning Tree Domains

The switch can be partitioned into multiple virtual bridges. Each virtual bridge can run an independent Spanning Tree instance. Each Spanning Tree instance is called a *Spanning Tree Domain* (STPD). Each STPD has its own root bridge and active path. After an STPD is created, one or more VLANs can be assigned to it.

A physical port can belong to multiple STPDs. In addition, a VLAN can span multiple STPDs.

The key points to remember when configuring VLANs and STP are:

- Each VLAN forms an independent broadcast domain.
- STP blocks paths to create a loop-free environment.
- Within any given STPD, all VLANs belonging to it use the same spanning tree.

To create an STPD, use the following command:

```
create stpd <stpd_name>
```

To delete an STPD, use the following command:

```
delete stpd <stpd_name>
```

For more detailed information about configuring STP and STP parameters, see “Configuring STP on the Switch” on page 159.

Member VLANs

When you add a VLAN to an STPD, that VLAN becomes a member of the STPD. The two types of member VLANs in an STPD are:

- Carrier
- Protected

Carrier VLAN

A carrier VLAN defines the scope of the STPD which includes the physical and logical ports that belong to the STPD and the 802.1Q tag used to transport EMISTP or PVST+ encapsulated BPDUs (see “Encapsulation Modes” on page 139 for more information about encapsulating STP BPDUs). Only one carrier VLAN can exist in a given STP domain although some of its ports can be outside the control of any STP domain at the same time.

The carrier VLAN’s StpdID must be identical to the VLANid of one of the member VLANs in that STP domain. See the section “Specifying the Carrier VLAN” on page 138, for an example.

Protected VLAN

Protected VLANs are all other VLANs that are members of the STP domain but do not define the scope of the STPD. These VLANs “piggyback” on the carrier VLAN. Protected VLANs do not transmit or receive STP BPDUs, but they are affected by STP state changes and inherit the state of the carrier VLAN. Protected VLANs can participate in multiple STP domains, but any particular port in the VLAN can belong to only one STP domain. Also known as non-carrier VLANs.

Specifying the Carrier VLAN

The following example:

- Creates and enables an STPD named *s8*
- Creates a VLAN named *v5*
- Assigns VLAN *v5* to STPD *s8*
- Creates the same tag ID for the VLAN and the STPD (the carrier VLAN's StpdID must be identical to the VLANid of one of the member VLANs)

```
create vlan v5
configure vlan v5 tag 100
configure vlan v5 add ports 1:1-1:20 tagged
create stpd s8
configure stpd s8 add vlan v5 ports all emistp
configure stpd s8 tag 100
enable stpd s8
```

Notice how the tag number for the VLAN *v5* and the STPD *s8* is identical (the tag is 100). By using identical tags, you have selected the carrier VLAN. The carrier VLAN's StpdID is identical to the VLANid of the member VLAN.

STPD Modes

An STPD has two modes of operation:

- 802.1d mode

  Use this mode for backward compatibility with previous STP versions and for compatibility with third-party switches using IEEE standard 802.1d. When configured in this mode, all rapid configuration mechanisms are disabled.

- 802.1w mode

  Use this mode for compatibility with Rapid Spanning Tree (RSTP). When configured in this mode, all rapid configuration mechanisms are enabled. The benefit of this mode is available on point-to-point links only and when the peer is likewise configured in 802.1w mode. If you do not select point-to-point links and the peer is not configured for 802.1w mode, the STPD fails back to 802.1d mode.

  RSTP is enabled or disabled on a per STPD basis only. You do not enable RSTP on a per port basis.

  For more information about RSTP and RSTP features, see "Rapid Spanning Tree Protocol" on page 148.

By default, the:

- STPD operates in 802.1d mode
- Default device configuration contains a single STPD called *s0*
- Default VLAN is a member of STPD s0 with autobind enabled

To configure the mode of operation of an STPD, use the following command:

```
configure stpd <stpd_name> mode [dot1d | dot1w]
```

All STP parameters default to the IEEE 802.1d values, as appropriate.

Encapsulation Modes

You can configure ports within an STPD to accept specific BPDU encapsulations. This STP port encapsulation is separate from the STP mode of operation. For example, you can configure a port to accept the PVST+ BPDU encapsulation while running in 802.1D mode. An STP port has three encapsulation modes:

- 802.1d mode

  This mode is used for backward compatibility with previous STP versions and for compatibility with third-party switches using IEEE standard 802.1d. BPDUs are sent untagged in 1D mode. Because of this, any given physical interface can have only *one* STPD running in 1D mode.

- Extreme Multiple Instance Spanning Tree Protocol (EMISTP) mode

  EMISTP mode is an extension of STP that allows a physical port to belong to multiple STPDs by assigning the port to multiple VLANs. EMISTP adds significant flexibility to STP network design. BPDUs are sent with an 802.1Q tag having an STPD instance Identifier (StpdID) in the VLANid field.

- PVST+ mode

  This mode implements PVST+ in compatibility with third-party switches running this version of STP. The STPDs running in this mode have a one-to-one relationship with VLANs, and send and process packets in PVST+ format.

These encapsulation modes are for STP ports, not for physical ports. When a physical port belongs to multiple STPDs, it is associated with multiple STP ports. It is possible for the physical port to run in different modes for different domains to which it belongs.

To configure the BPDU encapsulation mode for one or more STP ports, use the following command:

```
configure stpd <stpd_name> ports mode [dot1d | emistp | pvst-plus] <port_list>
```

To configure the default BPDU encapsulation mode on a per STPD basis, use the following command:

```
configure stpd <stpd_name> default-encapsulation [dot1d | emistp | pvst-plus]
```

Instead of accepting the default encapsulation modes of `dot1d` for the default STPD S0 and `emistp` for all other STPDs, this command allows you to specify the type of BPDU encapsulation to use for all ports added to the STPD (if not otherwise specified).

STPD Identifier

An StpdID is used to identify each STP domain. You assign the StpdID when configuring the domain, and that carrier VLAN cannot belong to another STPD.

An StpdID must be identical to the VLANid of one of the member VLANs in that STP domain.

NOTE

If an STPD contains at least one port not in 1D mode, the STPD must be configured with an StpdID.

STP States

Each port that belongs to a member VLAN participating in STP exists in one of the following states:

- Blocking

  A port in the blocking state does not accept ingress traffic, perform traffic forwarding, or learn MAC source addresses. The port does receive STP BPDUs. During initialization, the switch always enters the blocking state.

- Listening

  A port in the listening state does not accept ingress traffic, perform traffic forwarding, or learn MAC source addresses. The port does receive STP BPDUs. This is the first transitional state a port enters after being in the blocking state. The bridge listens for BPDUs from neighboring bridge(s) to determine whether the port should or should not be blocked.

- Learning

  A port in the learning state does not accept ingress traffic or perform traffic forwarding, but it begins to learn MAC source addresses. The port also receives and processes STP BPDUs. This is the second transitional state after listening. From learning, the port will change to either blocking or forwarding.

- Forwarding

  A port in the forwarding state accepts ingress traffic, learns new MAC source addresses, forwards traffic, and receives and processes STP BPDUs.

- Disabled

  A port in the disabled state does not participate in STP.

Binding Ports

The two ways to bind (add) ports to an STPD are: manually and automatically. By default, ports are manually added to an STPD.

Manually Binding Ports

To manually bind ports, use one of the following commands:

- `configure stpd <stpd_name> add vlan <vlan_name> ports [all | <port_list>] {[dot1d | emistp | pvst-plus]}`
- `configure vlan <vlan_name> add ports [all | <port_list>] stpd <stpd_name> {[dot1d | emistp | pvst-plus]}`

Both commands add all ports or a list of ports within a VLAN to a specified STPD provided the carrier VLAN already exists on the same set of ports. If the specified VLAN is not the carrier VLAN, and the specified ports are not bound to the carrier VLAN in the STPD, an error message is displayed.

NOTE

The carrier VLAN's StpdID must be identical to the VLANid of one of the member VLANs in that STP domain.

If you add a protected VLAN or port, it will inherit the carrier VLAN's encapsulation mode unless you specify the encapsulation mode when you execute the `configure stpd add vlan` or `configure vlan add ports stpd` commands. If you specify an encapsulation mode (`dot1d`, `emistp`, or `pvst-plus`), the

STP ports mode is changed to match, otherwise the STP ports inherit either the carrier VLANs encapsulation mode on that port or the STPD default encapsulation mode.

To remove ports, use the following command:

```
configure stpd <stpd_name> delete vlan <vlan_name> ports [all | <port_list>}
```

If you manually delete a protected VLAN or port, only that VLAN or port is removed. If you manually delete a carrier VLAN or port, all VLANs on that port (both carrier and protected) are deleted from that STPD.

To learn more about member VLANs, see "Member VLANs" on page 137. For more detailed information about these commands, see the *ExtremeWare XOS Command Reference Guide*.

Automatically Binding Ports

To automatically bind ports to an STPD when they are added to a VLAN, use the following command:

```
enable stpd <stpd_name> auto-bind vlan <vlan_name>
```

When you issue this command, any port or list of ports that you add to the carrier VLAN are automatically added to the STPD with autobind enabled. In addition, any port or list of ports that you remove from a carrier VLAN are automatically removed from the STPD. This allows the STPD to increase or decrease its span as ports are added to or removed from a carrier VLAN.

NOTE

The carrier VLAN's StpdID must be identical to the VLANid of one of the member VLANs in that STP domain.

Enabling autobind on a protected VLAN does not expand the boundary of the STPD. Protected VLANs are aware of STP state changes provided the same set of ports are members of the protected VLAN and the carrier VLAN. For example, assume you have the following scenario:

- Carrier VLAN named *v1*
- *v1* contains ports 3:1-3:2
- Protected VLAN named *v2*
- *v2* contains ports 3:1-3:4

Since *v1* contains ports 3:1-3:2, *v2* is only aware of the STP changes for ports 3:1 and 3:2, respectively. Ports 3:3 and 3:4 are not part of the STPD, which is why *v2* is not aware of any STP changes for those ports.

To remove ports, use the following command:

```
configure stpd <stpd_name> delete vlan <vlan_name> ports [all | <port_list>}
```

If you manually delete a port from the STPD on a VLAN that has been added by autobind, ExtremeWare XOS records the deletion so that the port does not get automatically added to the STPD after a system restart.

To learn more about the member VLANs, see "Member VLANs" on page 137. For more detailed information about these command, see the *ExtremeWare XOS Command Reference Guide*.

Rapid Root Failover

ExtremeWare XOS supports rapid root failover for faster STP failover recovery times in STP 802.1d mode. If the active root port link goes down ExtremeWare XOS recalculates STP and elects a new root port. Rapid root failover allows the new root port to immediately begin forwarding, skipping the standard listening and learning phases. Rapid root failover occurs only when the link goes down, and not when there is any other root port failure, such as missing BPDUs.

The default setting is disabled. To enable rapid root failover, use the following command:

```
enable stpd <stpd_name> rapid-root-failover
```

To display the configuration, use the following command:

```
show stpd {<stpd_name> | detail}
```

STP Configurations

When you assign VLANs to an STPD, pay careful attention to the STP configuration and its effect on the forwarding of VLAN traffic.

This section describes three types of STP configurations:

- Basic STP
- Multiple STPDs on a single port (EMISTP)
- A VLAN that spans multiple STPDs

Basic STP Configuration

This section describes a basic, 802.1D STP configuration. Figure 9 illustrates a network that uses VLAN tagging for trunk connections. The following four VLANs have been defined:

- *Sales* is defined on switch A, switch B, and switch M.
- *Personnel* is defined on switch A, switch B, and switch M.
- *Manufacturing* is defined on switch Y, switch Z, and switch M
- *Engineering* is defined on switch Y, switch Z, and switch M.
- *Marketing* is defined on all switches (switch A, switch B, switch Y, switch Z, and switch M).

Two STPDs are defined:

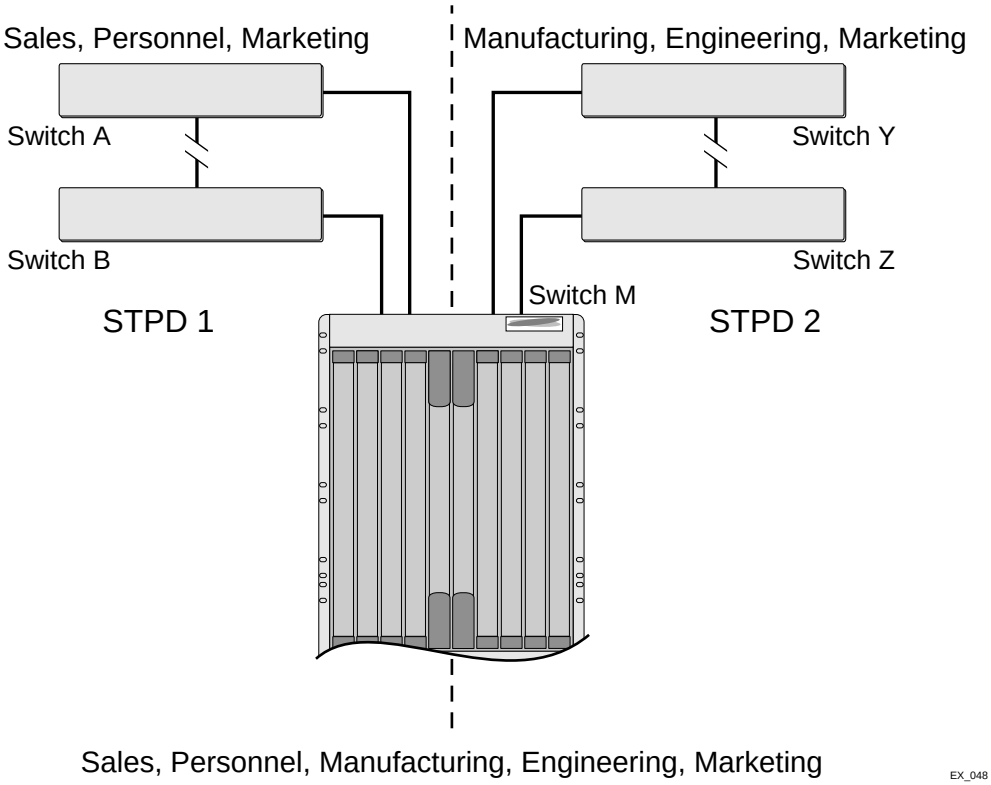
- STPD1 contains VLANs *Sales* and *Personnel.*
- STPD2 contains VLANs *Manufacturing* and *Engineering.*

The carrier and protected VLANs are also defined:

- *Sales* is the carrier VLAN on *STPD1.*
- *Personnel* is a protected VLAN on *STPD1.*
- *Manufacturing* is a protected VLAN on *STPD2.*

- *Engineering* is the carrier VLAN on *STPD2*.
- *Marketing* is a member of both STPD1 and STPD2 and is a protected VLAN.

Figure 9: Multiple Spanning Tree domains



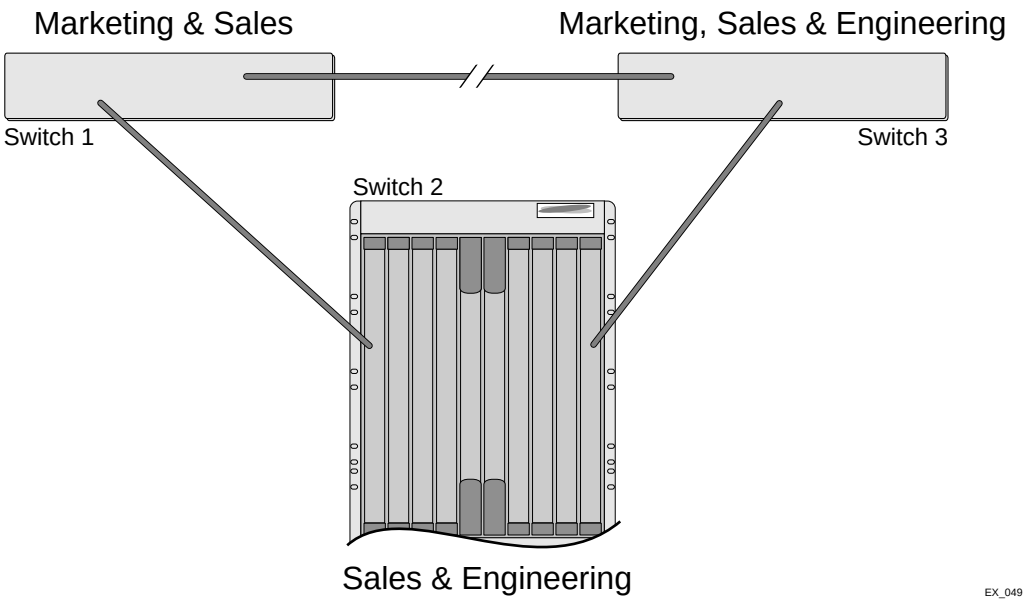
When the switches in this configuration start up, STP configures each STPD such that the topology contains no active loops. STP could configure the topology in a number of ways to make it loop-free.

In Figure 9, the connection between switch A and switch B is put into blocking state, and the connection between switch Y and switch Z is put into blocking state. After STP converges, all the VLANs can communicate, and all bridging loops are prevented.

The protected VLAN *Marketing*, which has been assigned to both STPD1 and STPD2, communicates using all five switches. The topology has no loops, because STP has already blocked the port connection between switch A and switch B, and between switch Y and switch Z.

Within a single STPD, you must be extra careful when configuring your VLANs. Figure 10 illustrates a network that has been incorrectly set up using a single STPD so that the STP configuration disables the ability of the switches to forward VLAN traffic.

Figure 10: Tag-based STP configuration



The tag-based network in Figure 10 has the following configuration:

- Switch 1 contains VLAN *Marketing* and VLAN *Sales.*
- Switch 2 contains VLAN *Engineering* and VLAN *Sales.*
- Switch 3 contains VLAN *Marketing,* VLAN *Engineering,* and VLAN *Sales.*
- The tagged trunk connections for three switches form a triangular loop that is not permitted in an STP topology.
- All VLANs in each switch are members of the same STPD.

STP can block traffic between switch 1 and switch 3 by disabling the trunk ports for that connection on each switch.

Switch 2 has no ports assigned to VLAN marketing. Therefore, if the trunk for VLAN *Marketing* on switches 1 and 3 is blocked, the traffic for VLAN *Marketing* will not be able to traverse the switches.

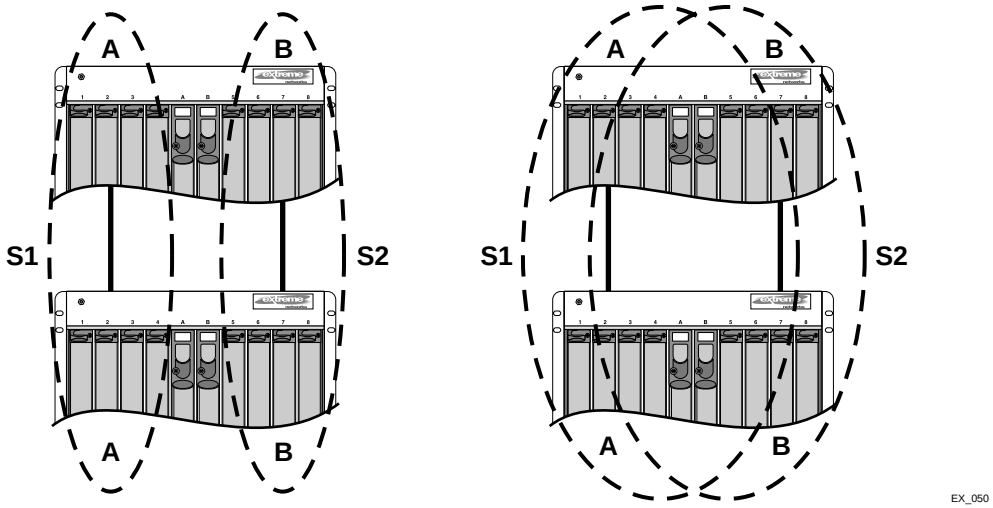
NOTE

If an STPD contains multiple VLANs, all VLANs should be configured on all ports in that domain, except for ports that connect to hosts (edge ports).

Multiple STPDs on a Port

Traditional 802.1d STP has some inherent limitations when addressing networks that have multiple VLANs and multiple STPDs. For example, consider the sample depicted in Figure 11.

Figure 11: Limitations of traditional STPD



The two switches are connected by a pair of parallel links. Both switches run two VLANs, A and B. To achieve load-balancing between the two links using the traditional approach, you would have to associate A and B with two different STPDs, called S1 and S2, respectively, and make the left link carry VLAN A traffic while the right link carry VLAN B traffic (or vice versa). If the right link fails, S2 is broken and VLAN B traffic is disrupted.

To optimize the solution, you can use the Extreme Multiple Instance Spanning (EMISTP) mode, which allows a port to belong to multiple STPDs. EMISTP adds significant flexibility to STP network design. Referring to Figure 11, using EMISTP, you can configure all four ports to belong to both VLANs.

Assuming that S1 and S2 still correspond to VLANs A and B, respectively, you can fine-tune STP parameters to make the left link active in S1 and blocking in S2, while the right link is active in S2 and blocking in S1. Once again, if the right link fails, the left link is elected active by the STP algorithm for S2, without affecting normal switching of data traffic.

Using EMISTP, an STPD becomes more of an abstract concept. It does not necessarily correspond to a physical domain. It is better regarded as a vehicle to carry VLANs that have STP instances. Because VLANs can overlap, so do STPDs. However, even if the different STPDs share the entire topology or part of the redundant topology, the STPDs react to topology change events in an independent fashion.

VLAN Spanning Multiple STPDs

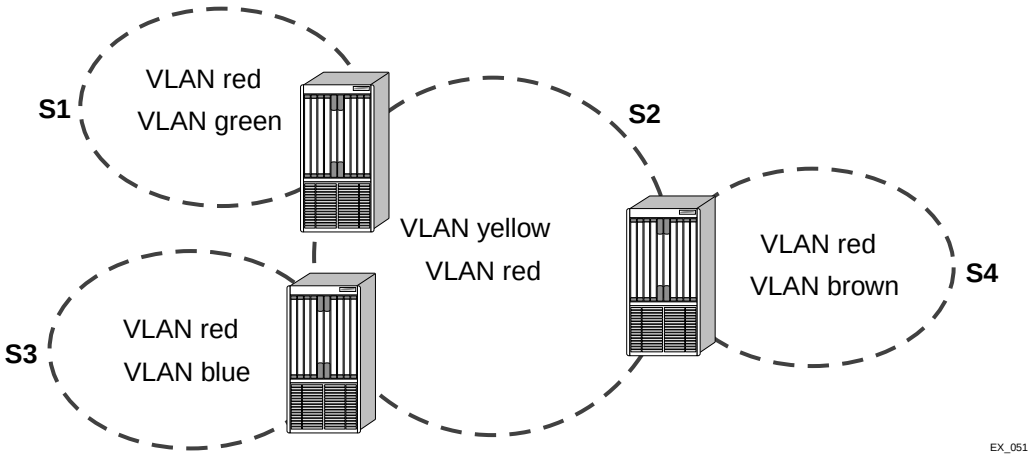
Traditionally, the mapping from VLANs to STP instances have been one-to-one, or many-to-one. In both cases, a VLAN is wholly contained in a single instance. In practical deployment there are cases in which a one-to-many mapping is desirable. In a typical large enterprise network, for example, VLANs span multiple sites and/or buildings. Each site represents a redundant looped area. However, between any two sites the topology is usually very simple.

Alternatively, the same VLAN may span multiple large geographical areas (because they belong to the same enterprise) and may traverse a great many nodes. In this case, it is desirable to have multiple STP domains operating in a single VLAN, one for each looped area. The justifications include the following:

- The complexity of the STP algorithm increases, and performance drops, with the size and complexity of the network. The 802.1d standard specifies a maximum network diameter of seven hops. By segregating a big VLAN into multiple STPDs, you reduce complexity and enhance performance.
- Local to each site, there may be other smaller VLANs that share the same redundant looped area with the large VLAN. Some STPDs must be created to protect those VLAN. The ability to partition VLANs allows the large VLAN to be “piggybacked” in those STPDs in a site-specific fashion.

Figure 12 has five domains. VLANs green, blue, brown, and yellow are local to each domain. VLAN red spans all of the four domains. Using a VLAN that spans multiple STPDS, you do not have to create a separate domain for VLAN red. Instead, VLAN red is “piggybacked” onto those domains local to other VLANs.

Figure 12: VLAN spanning multiple STPDs



In addition, the configuration in Figure 12 has these features:

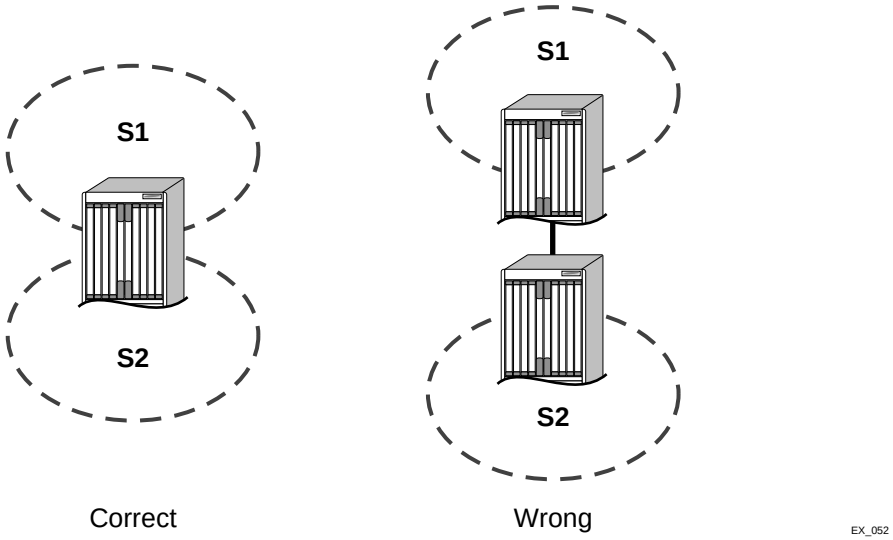
- Each site can be administered by a different organization or department within the enterprise. Having a site-specific STP implementation makes the administration more flexible and convenient.
- Between the sites the connection usually traverse distribution switches in ways that are known beforehand to be “safe” with STP. In other words, the looped areas are already well-defined.

EMISTP Deployment Constraints

While EMISTP greatly enhances STP capability, these features must deployed with care. This section discusses configuration issues that, if not followed, could lead to an improper deployment of EMISTP. This section also provides the restrictive principles to abide by in network design.

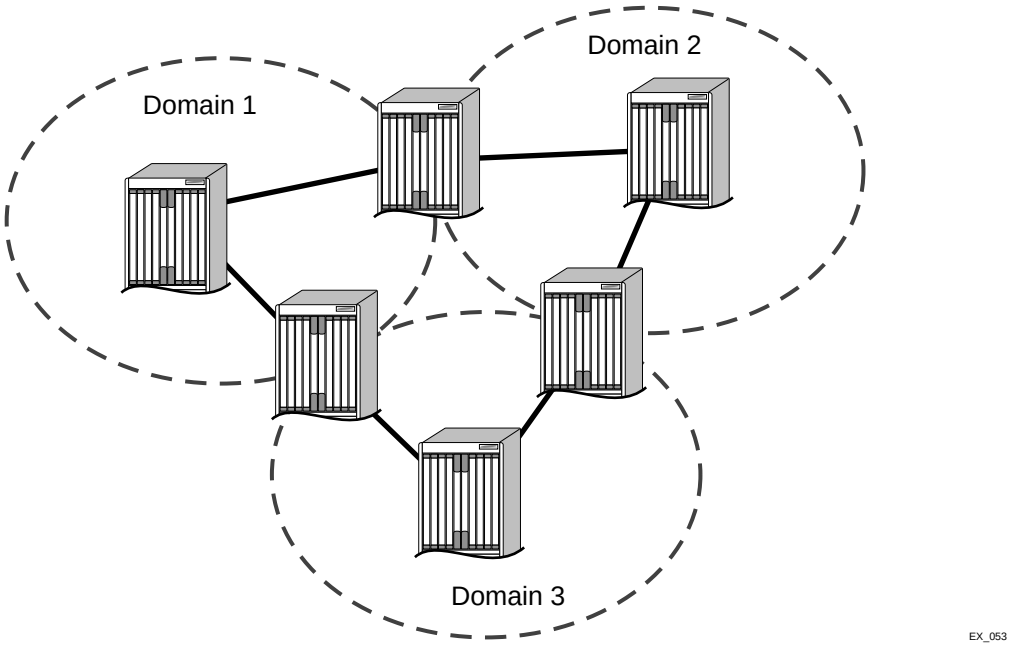
- While a physical port can belong to multiple STPDs, any VLAN on that port can be in only one domain. Put another way, a VLAN can not belong to two domains on the same physical port.
- While a VLAN can span multiple domains, any LAN segment in that VLAN must be in the same STPD. VLANs traverse domains only inside switches, not across links. On a single switch, however, bridge ports for the same VLAN can be assigned to different STPDs. This scenario is illustrated in Figure 13.

Figure 13: VLANs traverse domains inside switches



- The VLAN partition feature is deployed under the premise that the overall inter-domain topology for that VLAN is loop-free. Consider the case in Figure 14, VLAN red (the only VLAN in the figure) spans domains 1, 2, and 3. Inside each domain, STP produces a loop-free topology. However, VLAN red is still looped, because the three domains form a ring among themselves.

Figure 14: Looped VLAN topology



A necessary (but not sufficient) condition for a loop-free inter-domain topology is that every two domains only meet at a single crossing point.

Per-VLAN Spanning Tree

Switching products that implement Per-VLAN Spanning Tree (PVST) have been in existence for many years and are widely deployed. To support STP configurations that use PVST, ExtremeWare XOS has an operational mode called PVST+.

NOTE

In this document, PVST and PVST+ are used interchangeably. PVST+ is an enhanced version of PVST that is interoperable with 802.1Q STP. The following discussions are in regard to PVST+, if not specifically mentioned.

STPD VLAN Mapping

Each VLAN participating in PVST+ must be in a separate STPD and the VLAN number must be the same as the STPD identifier (StpdID). As a result, PVST+ protected VLANs cannot be partitioned.

This fact does not exclude other non-PVST+ protected VLANs from being grouped into the same STPD. A protected PVST+ VLAN can be joined by multiple non-PVST+ protected VLANs to be in the same STP domain.

Native VLAN

In PVST+, the native VLAN must be peered with default VLAN on Extreme devices, as both are the only VLAN allowed to send and receive untagged packets on the physical port.

Third-party PVST+ devices send VLAN 1 packets in a special manner. ExtremeWare XOS does not support PVST+ for VLAN 1. Therefore, when the switch receives a packet for VLAN 1, the packet is dropped.

When a PVST+ instance is disabled, the fact that PVST+ uses a different packet format raises an issue. If the STPD also contains ports not in PVST+ mode, the flooded packet has an incompatible format with those ports. The packet is not recognized by the devices connected to those ports.

Rapid Spanning Tree Protocol

The Rapid Spanning Tree Protocol (RSTP; 802.1w) provides an enhanced spanning tree algorithm that improves the convergence speed of bridged networks. RSTP takes advantage of point-to-point links in the network and actively confirms that a port can safely transition to the forwarding state without relying on any timer configurations. If a network topology change or failure occurs, RSTP rapidly recovers network connectivity by confirming the change locally before propagating that change to other devices across the network. For broadcast links, there is no difference in convergence time between STP and RSTP.

RSTP supersedes legacy STP protocols, supports the existing STP parameters and configurations, and allows for seamless interoperability with legacy STP.

RSTP Terms

Table 28 describes the terms associated with RSTP.

Table 28: RSTP terms

Term	Description
root port	Provides the shortest path to the root bridge. All bridges except the root bridge, contain one root port. For more information about the root port, see “Port Roles” on page 149.
designated port	Provides the shortest path connection to the root bridge for the attached LAN segment. Each LAN segment has only one designated port. For more information about the designated port, see “Port Roles” on page 149.
alternate port	Supplies an alternate path to the root bridge and the root port. For more information about the alternate port, see “Port Roles” on page 149.
backup port	Supports the designated port on the same attached LAN segment. Backup ports only exist when the bridge is connected as a self-loop or to a shared-media segment. For more information about the backup port, see “Port Roles” on page 149.
edge ports	Ports that connect to non-STP devices such as routers, endstations, and other hosts.
root bridge	The bridge with the best bridge identifier selected to be the root bridge. The network has only one root bridge. The root bridge is the only bridge in the network that does not have a root port.

RSTP Concepts

This section describes important RSTP concepts.

Port Roles

RSTP uses information from BPDUs to assign port roles for each LAN segment. Port roles are not user-configurable. Port role assignments are determined based on the following criteria:

- A unique bridge identifier (MAC address) associated with each bridge
- The path cost associated with each bridge port
- A port identifier associated with each bridge port

RSTP assigns one of four port roles to bridge ports in the network, as described in Table 29.

Table 29: RSTP port roles

Port Role	Description
Root	Provides the shortest path to the root bridge. Each bridge has only one root port; the root bridge does not have a root port. If a bridge has two or more ports with the same path cost, the port with the best port identifier becomes the root port.
Designated	Provides the shortest path connection to the root bridge for the attached LAN segment. To prevent loops in the network, there is only one designated port on each LAN segment. To select the designated port, all bridges that are connected to a particular segment listen to each other’s BPDUs and agree on the bridge sending the best BPDU. The corresponding port on that bridge becomes the designated port. If there are two or more ports connected to the LAN, the port with the best port identifier (lowest MAC address) becomes the designated port.
Alternate	Provides an alternate path to the root bridge and the root port.

Table 29: RSTP port roles (continued)

Port Role	Description
Backup	Supports the designated port on the same attached LAN segment. Backup ports only exist when the bridge is connected as a self-loop or to a shared-media segment.

When RSTP stabilizes, all:

- Root ports and designated ports are in the forwarding state
- Alternate ports and backup ports are in the blocking state

RSTP makes the distinction between the alternate and backup port roles to describe the rapid transition of the alternate port to the forwarding state if the root port fails.

Ports that connect to non-STP devices are edge ports. Edge ports do not participate in RSTP, and their role is not confirmed. Edge ports immediately enter the forwarding state.

Link Types

You can configure the link type of a port in an STPD. RSTP tries to rapidly move designated point-to-point links into the forwarding state when a network topology change or failure occurs. For rapid convergence to occur, the port must be configured as a point-to-point link.

Table 30 describes the link types.

Table 30: RSTP link types

Port Role	Description
Auto	Specifies the switch to automatically determine the port link type. An auto link behaves like a point-to-point link if the link is in full duplex mode or if link aggregation is enabled on the port. Otherwise, the link behaves like a broadcast link used for 802.1w configurations.
Edge	Specifies a port that does not have a bridge attached. An edge port is placed and held in the STP forwarding state unless a BPDU is received by the port.
Broadcast	Specifies a port attached to a LAN segment with more than two bridges. A port with a broadcast link type cannot participate in rapid reconfiguration. By default, all ports are broadcast links.
Point-to-point	Specifies a port attached to a LAN segment with only two bridges. A port with port-to-port link type can participate in rapid reconfiguration. Used for 802.1w configurations.

Configuring Link Types. By default, all ports are broadcast links. To configure the ports in an STPD, use the following command:

```
configure stpd <stpd_name> ports link-type [auto | edge | broadcast | point-to-point]
<port_list>
```

- `auto`—Configures the ports as auto links. If the link is in full duplex mode, or if link aggregation is enabled on the port, an auto link behaves like a point-to-point link.
- `edge`—Configures the ports as edge ports.
- `point-to-point`—Configures the ports for an RSTP environment.

To change the existing configuration of a port in an STPD, and return the port to factory defaults, use the following command:

```
unconfigure stpd <stpd_name> ports link-type <port_list>
```

To display detailed information about the ports in an STPD, use the following command:

```
show stpd <stpd_name> ports {<port_list> {detail}}
```

RSTP Timers

For RSTP to rapidly recover network connectivity, RSTP requires timer expiration. RSTP derives many of the timer values from the existing configured STP timers to meet its rapid recovery requirements rather than relying on additional timer configurations. Table 31 describes the user-configurable timers, and Table 32 describes the timers that are derived from other timers and not user-configurable.

Table 31: User-configurable timers

Timer	Description
Hello	The root bridge uses the hello timer to send out configuration BPDUs through all of its forwarding ports at a predetermined, regular time interval. The default value is 2 seconds. The range is 1 to 10 seconds.
Forward delay	A port moving from the blocking state to the forwarding state uses the forward delay timer to transition through the listening and learning states. In RSTP, this timer complements the rapid configuration behavior. If none of the rapid rules are in effect, the port uses legacy STP rules to move to the forwarding state. The default is 15 seconds. The range is 4 to 30 seconds.

Table 32: Derived timers

Timer	Description
TCN	The root port uses the TCN timer when it detects a change in the network topology. The TCN timer stops when the topology change timer expires or upon receipt of a topology change acknowledgement. The default value is the same as the value for the bridge hello timer.
Topology Change	The topology change timer determines the total time it takes the forwarding ports to send configuration BPDUs. The default value for the topology change timer depends upon the mode of the port. • 1d mode—The sum of the forward delay timer (default value is 15 seconds; range of 4 to 30 seconds) and the max age timer (default value is 20 seconds; range of 6 to 40 seconds). • 1w mode—Double the hello timer (default value is 4 seconds)
Message age	A port uses the message age timer to time out receiving BPDUs. When a port receives a superior or equal BPDU, the timer restarts. When the timer expires, the port becomes a designated port and a configuration update occurs. If the bridge operates in 1w mode and receives an inferior BPDU, the timer expires early. The default value is the same as the STPD bridge max age parameter.
Hold	A port uses the hold timer to restrict the rate that successive BPDUs can be sent. The default value is the same as the value for the bridge hello timer.
Recent backup	The timer starts when a port leaves the backup role. When this timer is running, the port cannot become a root port. The default value is double the hello time (4 seconds).

Table 32: Derived timers (continued)

Timer	Description
Recent root	The timer starts when a port leaves the root port role. When this timer is running, another port cannot become a root port unless the associated port is put into the blocking state. The default value is the same as the forward delay time.

The protocol migration timer is neither user-configurable nor derived; it has a set value of 3 seconds. The timer starts when a port transitions from STP (802.1d) mode to RSTP (802.1w) mode and vice versa. This timer must expire before further mode transitions can occur.

RSTP Operation

In an RSTP environment, a point-to-point link LAN segment has two bridges. A switch that considers itself the unique, designated bridge for the attached LAN segment sends a “propose” message to the other bridge to request a confirmation of its role. The other bridge on that LAN segment replies with an “agree” message if they agree with the proposal. The receiving bridge immediately moves its designated port into the forwarding state.

Before a bridge replies with an “agree” message, it reverts all of its designated ports into the blocking state. This introduces a temporary partition into the network. The bridge then sends another “propose” message on all of its designated ports for further confirmation. Since all of the connections are blocked, the bridge immediately sends an “agree” message to unblock the proposing port without having to wait for further confirmations to come back or without the worry of temporary loops.

Beginning with the root bridge, each bridge in the network engages in the exchange of “propose” and “agree” messages until they reach the edge ports. Edge ports connect to non-STP devices and do not participate in RSTP. Their role does not need to be confirmed. If an edge port receives a BPDU, it enters an inconsistency state. An inconsistency state puts the edge port into the blocking state and starts the message age timer. Every time the edge port receives a BPDU, the message age timer restarts. The edge port remains in the blocking state until no further BPDUs are received and the message age timer expires.

RSTP attempts to transition root ports and designated ports to the forwarding state and alternate ports and backup ports to the blocking state as rapidly as possible.

A port transitions to the forwarding state if any of the following is true. The port:

- Has been in either a root or designated port role long enough that the spanning tree information supporting this role assignment has reached all of the bridges in the network.

  NOTE

  RSTP is backward compatible with STP, so if a port does not move to the forwarding state with any of the RSTP rapid transition rules, a forward delay timer starts and STP behavior takes over.

- Is now a root port and no other ports have a recent role assignment that contradicts with its root port role.
- Is a designated port and attaches to another bridge by a point-to-point link and receives an “agree” message from the other bridge port.
- Is an edge port.

  An edge port is a port connected to a non-STP device and is in the forwarding state.

The following sections provide more information about RSTP behavior.

Root Port Rapid Behavior

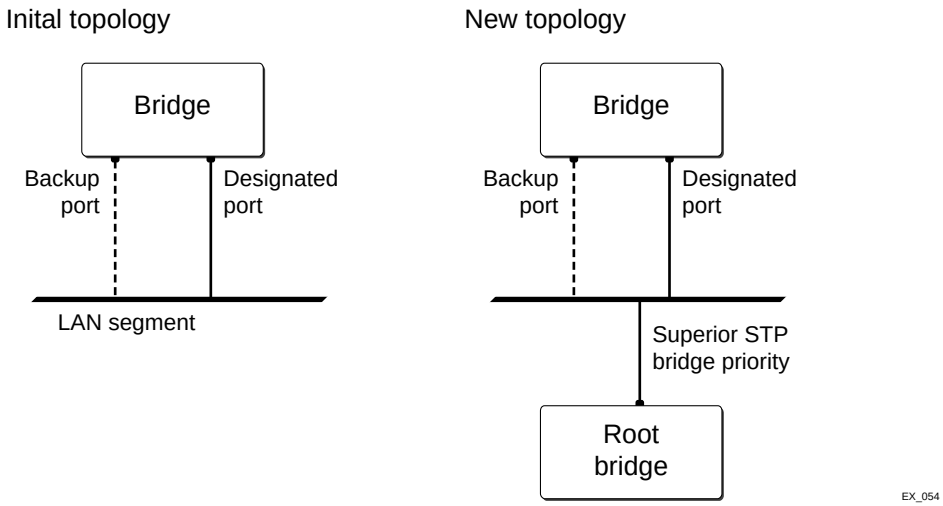
In Figure 15, the diagram on the left displays the initial network topology with a single bridge having the following:

- Two ports connected to a shared LAN segment
- One port is the designated port
- One port is the backup port

The diagram on the right displays a new bridge that:

- Is connected to the LAN segment
- Has a superior STP bridge priority
- Becomes the root bridge and sends a BPDU to the LAN that is received by both ports on the old bridge

Figure 15: Example of root port rapid behavior



If the backup port receives the BPDU first, STP processes this packet and temporarily elects this port as the new root port while the designated port's role remains unchanged. If the new root port is immediately put into the forwarding state, there is a loop between these two ports.

To prevent this type of loop from occurring, the recent backup timer starts. The root port transition rule does not allow a new root port to be in the forwarding state until the recent backup timer expires.

Another situation may arise if you have more than one bridge, and you lower the port cost for the alternate port which makes it the new root port. The previous root port is now an alternate port. Depending on your STP implementation, STP may set the new root port to the forwarding state before setting the alternate port to the blocking state. This may cause a loop.

To prevent this type of loop from occurring, the recent root timer starts when the port leaves the root port role. The timer stops if the port enters the blocking state. RSTP requires that the recent root timer stops on the previous root port before the new root port can enter the forwarding state.

Designated Port Rapid Behavior

When a port becomes a new designated port, or the STP priority changes on an existing designated port, the port becomes an *unsynced* designated port. In order for an unsynced designated port to rapidly move into the forwarding state, the port must propose a confirmation of its role on the attached LAN segment, unless the port is an edge port. Upon receiving an “agree” message, the port immediately enters the forwarding state.

If the receiving bridge does not agree and it has a superior STP priority, the receiving bridge replies with its own BPDU. Otherwise, the receiving bridge keeps silent and the proposing port enters the forwarding state and starts the forward delay timer.

The link between the new designated port and the LAN segment must be a point-to-point link. If there is a multi-access link, the “propose” message is sent to multiple recipients. If only one of the recipients agrees with the proposal, it is possible for the port to erroneously enter the forwarding state after receiving a single “agree” message.

Receiving Bridge Behavior

The receiving bridge must decide whether or not to accept a proposal from a port. Upon receiving a proposal for a root port, the receiving bridge:

- Processes the BPDU and computes the new STP topology
- Synchronizes all of the designated ports if the receiving port is the root port of the new topology
- Puts all unsynced, designated ports into the blocking state
- Sends down further “propose” messages
- Sends back an “agree” message through the root port

If the receiving bridge receives a proposal for a designated port, the bridge replies with its own BPDU. If the proposal is for an alternate or backup port, the bridge keeps silent.

Propagating Topology Change Information

When a change occurs in the topology of the network, such events are communicated through the network.

In an RSTP environment, only non-edge ports entering the forwarding state cause a topology change. A loss of network connectivity is not considered a topology change; however, a gain in network connectivity needs to be communicated. When an RSTP bridge detects a topology change, it starts the topology change timer, sets the topology change flag on its BPDUs, floods all of the forwarding ports in the network (including the root ports), and flushes the learned MAC address entries.

Rapid Reconvergence

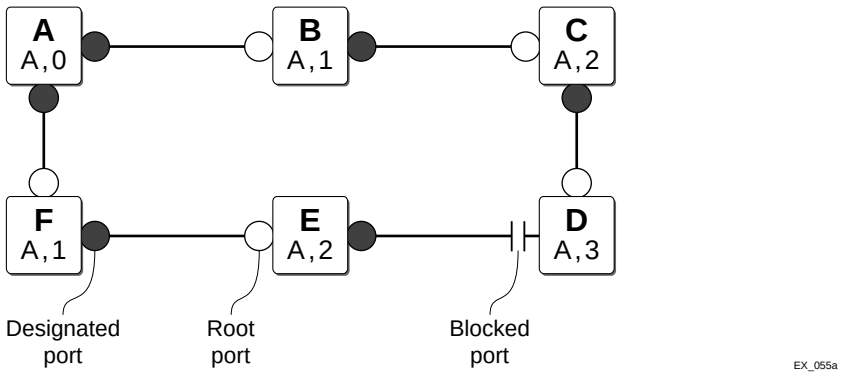
This section describes the RSTP rapid behavior following a topology change. In this example, the bridge priorities are assigned based on the order of their alphabetical letters; bridge A has a higher priority than bridge F.

Suppose we have a network, as shown in Figure 16, with six bridges (bridge A through bridge F) where the following is true:

- Bridge A is the root bridge.
- Bridge D contains an alternate port in the blocking state.

- All other ports in the network are in the forwarding state.

Figure 16: Initial network configuration



The following steps describe how the network reconverges.

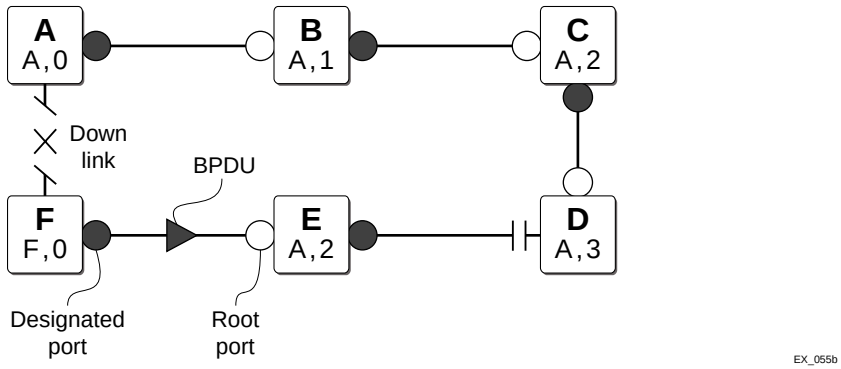
1 If the link between bridge A and bridge F goes down, bridge F detects the root port is down. At this point, bridge F:

- Immediately disables that port from the STP
- Performs a configuration update

After the configuration update, bridge F:

- Considers itself the new root bridge
- Sends a BPDU message on its designated port to bridge E

Figure 17: Down link detected



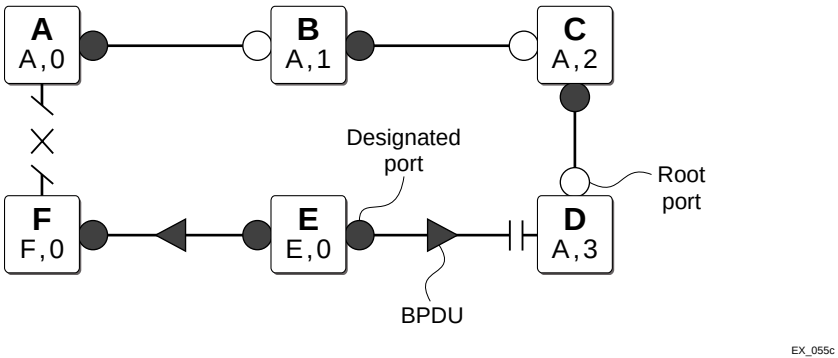
2 Bridge E believes that bridge A is the root bridge. When bridge E receives the BPDU on its root port from bridge F, bridge E:

- Determines that it received an inferior BPDU
- Immediately begins the max age timer on its root port
- Performs a configuration update

After the configuration update, bridge E:

- Regards itself as the new root bridge
- Sends BPDU messages on both of its designated ports to bridges F and D, respectively

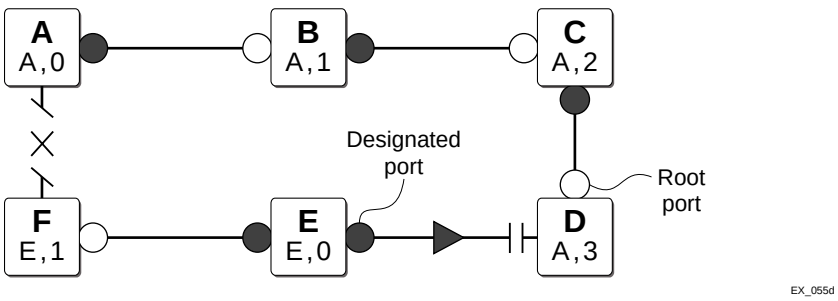
Figure 18: New root bridge selected



3 When bridge F receives the superior BPDU and configuration update from bridge E, bridge F:

- Decides that the receiving port is the root port
- Determines that bridge E is the root bridge

Figure 19: Communicating new root bridge status to neighbors



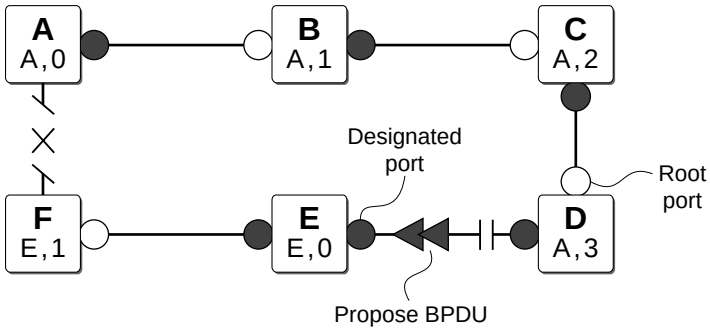
4 Bridge D believes that bridge A is the root bridge. When bridge D receives the BPDU from bridge E on its alternate port, bridge D:

- Immediately begins the max age timer on its alternate port
- Performs a configuration update

After the configuration update, bridge D:

- Moves the alternate port to a designated port
- Sends a “propose” message to bridge E to solicit confirmation of its designated role and to rapidly move the port into the designated state

Figure 20: Sending a propose message to confirm a port role



5 Upon receiving the proposal, bridge E:

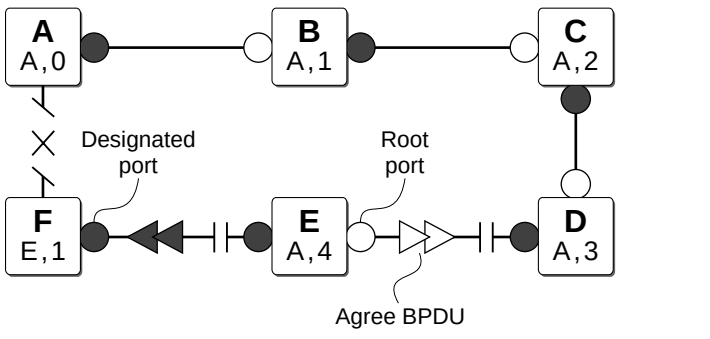
- Performs a configuration update
- Changes its receiving port to a root port

  The existing designated port enters the blocking state.

Bridge E then sends:

- A “propose” message to bridge F
- An “agree” message from its root port to bridge D

Figure 21: Communicating port status to neighbors



6 To complete the topology change:

- Bridge D moves the port that received the agree message into the forwarding state.
- Bridge F confirms that its receiving port (the port that received the “propose” message) is the root port, and immediately replies with an “agree” message to bridge E to unblock the proposing port.

Figure 22: Completing the topology change

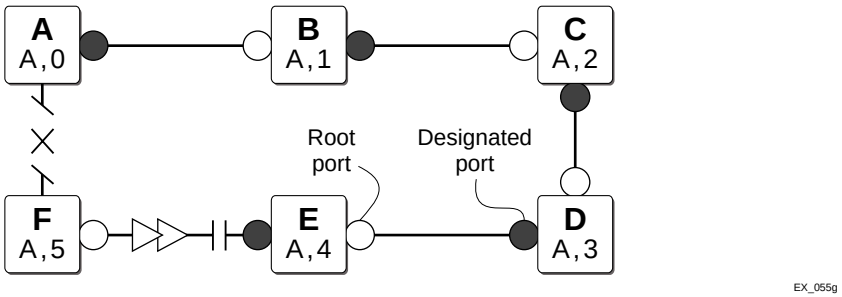
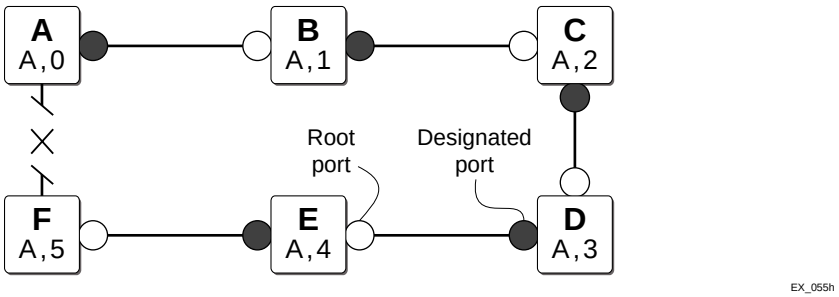


Figure 23 displays the new topology.

Figure 23: Final network configuration



Compatibility With STP (802.1d)

RSTP interoperates with legacy STP protocols; however, the rapid convergence benefits are lost when interacting with legacy STP bridges.

Each RSTP bridge contains a port protocol migration state machine to ensure that the ports in the STPD operate in the correct, configured mode. The state machine is a protocol entity within each bridge configured to run in 802.1w mode. For example, a compatibility issue occurs if you configure 802.1w mode and the bridge receives an 802.1d BPDU on a port. The receiving port starts the protocol migration timer and remains in 802.1d mode until the bridge stops receiving 802.1d BPDUs. Each time the bridge receives an 802.1d BPDU, the timer restarts. When the port migration timer expires, no more 802.1d BPDUs have been received and the bridge returns to its configured setting, 802.1w mode.

STP Rules and Restrictions

This section summarizes the rules and restrictions for configuring STP.

- The carrier VLAN must span all of the ports of the STPD.
- The StpdID must be the VLANid of one of its member VLANs, and that VLAN cannot be partitioned.
- A default VLAN can not be partitioned. If a VLAN traverses multiple STP domains, the VLAN must be tagged.
- An STPD can carry, at most, one VLAN running in PVST+ mode, and its StpdID must be identical with that VLANid. In addition, the PVST+ VLAN cannot be partitioned.
- The default VLAN of a PVST+ port must be identical with the native VLAN on the PVST+ device connected to that port.
- If a port supports 802.1d-STPD, then the port must be configured with a default VLAN. If not, the BPDUs for that STPD are not flooded when the STPD is disabled.
- If an STPD contains both PVST+ and non-PVST+ ports, it must be enabled. If it is disabled, the BPDUs are flooded in the format of the incoming STP port, which may be incompatible with those of the connected devices.
- 802.1d ports must be untagged; EMISTP/PVST+ ports must be tagged.
- An STPD with multiple VLANs must contain only VLANs that belong to the same virtual router instance.

Configuring STP on the Switch

To configure basic STP:

1 Create one or more STP domains using the following command:

```
create stpd <stpd_name>
```

NOTE

STPD, VLAN, and QoS profile names must all be unique. For example, a name used to identify a VLAN cannot be used when you create an STPD or a QoS profile.

2 Add one or more VLANs to the STPD using the following command:

```
configure stpd <stpd_name> add vlan <vlan_name> ports [all | <port_list>] {[dot1d
| emistp | pvst-plus]}
```

3 Define the carrier VLAN using the following command:

```
configure stpd <stpd_name> tag <stpd_tag>
```

NOTE

The carrier VLAN's StpdID must be identical to the VLANid of one of the member VLANs.

4 Enable STP for one or more STP domains using the following command:

```
enable stpd {<stpd_name>}
```

After you have created the STPD, you can optionally configure STP parameters for the STPD.

NOTE

You should not configure any STP parameters unless you have considerable knowledge and experience with STP. The default STP parameters are adequate for most networks.

The following parameters can be configured on each STPD:

- Hello time
- Forward delay
- Max age
- Bridge priority
- StpdID

The following parameters can be configured on each port:

- Path cost
- Port priority
- Port mode

NOTE

The device supports the RFC 1493 Bridge MIB, RSTP-03, and Extreme Networks STP MIB. Parameters of the s0 default STPD support RFC 1493 and RSTP-03. Parameters of any other STPD support the Extreme Networks STP MIB.

NOTE

If an STPD contains at least one port not in dot1D mode, the STPD must be configured with an StpdID.

STP Configuration Examples

This section provides three configuration examples:

- Basic 802.1d STP
- EMISTP
- RSTP 802.1w

Basic 802.1d Configuration Example

The following example:

- Creates the VLAN *Engineering*
- Configures the VLANid
- Adds ports to the VLAN *Engineering*
- Creates an STPD named *Backbone_st*
- Enables autobind to automatically add or remove ports from the STPD

- Assigns the *Engineering* VLAN to the STPD
- Assigns the carrier VLAN
- Enables STP

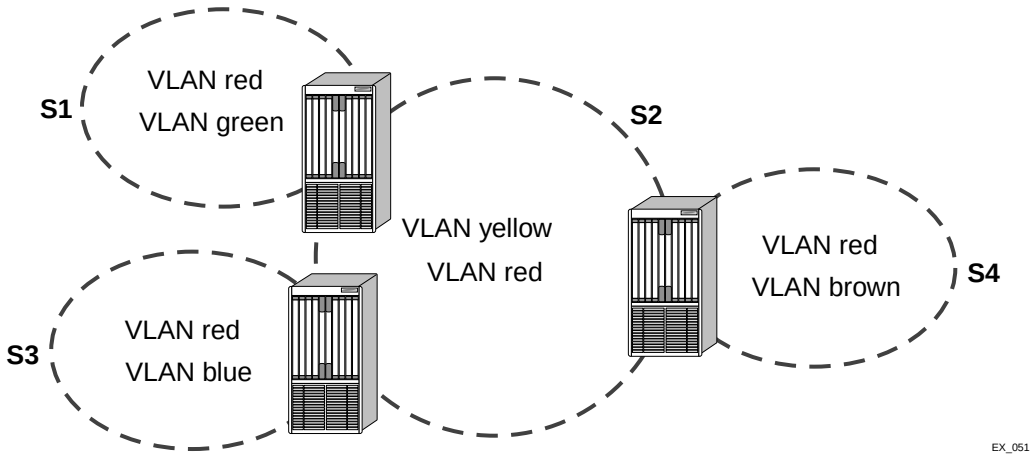
```
create vlan engineering
configure vlan engineering tag 150
configure vlan engineering add ports 2:5-2:10 tagged

create stpd backbone_st
enable stpd backbone_st auto-bind vlan engineering
configure stpd backbone_st tag 150
enable stpd backbone_st
```

By default, the port encapsulation mode for user-defined STPDs is `emistp`.

EMISTP Configuration Example

Figure 24 is an example of EMISTP.

Figure 24: EMISTP configuration example



The following commands configure the switch located between S1 and S2:

NOTE

By default, all ports added to a user-defined STPD are in `emistp` *mode, unless otherwise specified.*

```
create vlan red
configure red tag 100
configure red add ports 1:1-1:4 tagged

create vlan green
configure green tag 200
configure green add ports 1:1-1:2 tagged

create vlan yellow
configure yellow tag 300
configure yellow add ports 1:3-1:4 tagged
```

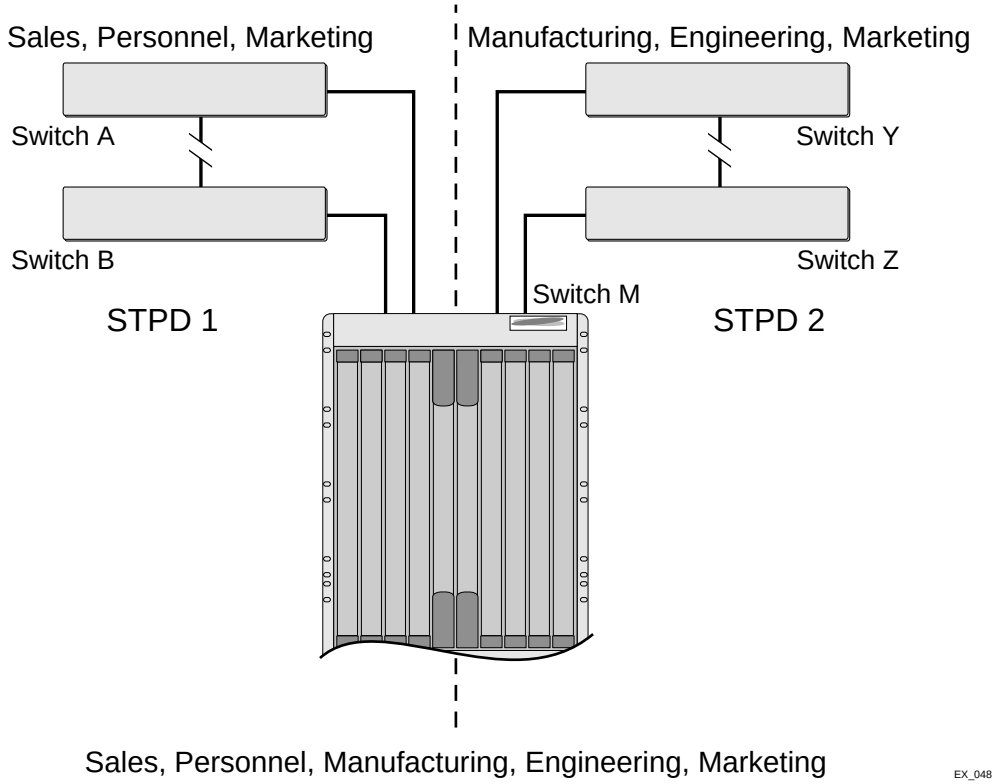
```
create stpd s1
configure stpd s1 add green ports all
configure stpd s1 tag 200
configure stpd s1 add red ports 1:1-1:2 emistp
enable stpd s1

create stpd s2
configure stpd s2 add yellow ports all
configure stpd s2 tag 300
configure stpd s2 add red ports 1:3-1:4 emistp
enable stpd s2
```

RSTP 802.1w Configuration Example

Figure 25 is an example of a network with multiple STPDs that can benefit from RSTP. For RSTP to work, you need to do the following:

- Create an STPD.
- Configure the mode of operation for the STPD.
- Create the VLANs and assign the VLANid and the ports.
- Assign the carrier VLAN.
- Add the VLANs to the STPD.
- Configure the port link types.
- Enable STP.

Figure 25: RSTP example



In this example, the commands configure switch A in STPD1 for rapid reconvergence. Use the same commands to configure each switch and STPD in the network.

```
create stpd stpd1
configure stpd stpd1 mode dot1w

create vlan sales
create vlan personnel
create vlan marketing
configure vlan sales tag 100
configure vlan personnel tag 200
configure vlan marketing tag 300
configure vlan sales add ports 1:1,2:1 tagged
configure vlan personnel add ports 1:1,2:1 tagged
configure vlan marketing add ports 1:1,2:1 tagged

configure stpd stpd1 add vlan sales ports all
configure stpd stpd1 add vlan personnel ports all
configure stpd stpd1 add vlan marketing ports all

configure stpd stpd1 ports link-type point-to-point 1:1,2:1

configure stpd stpd1 tag 100

enable stpd stpd1
```

Displaying STP Settings

To display STP settings, use the following command:

```
show stpd {<stpd_name> | detail}
```

This command displays the following information:

- STPD name
- STPD state
- STPD mode of operation
- Rapid Root Failover
- Tag
- Ports
- Active VLANs
- Bridge Priority
- Bridge ID
- Designated root
- STPD configuration information

To display the STP state of a port, use the following command:

```
show stpd <stpd_name> ports {<port_list> {detail}}
```

This command displays the following information:

- STPD port configuration
- STPD port mode of operation
- STPD path cost
- STPD priority
- STPD state (root bridge, and so on)
- Port role (root bridge, edge port, etc.)
- STPD port state (forwarding, blocking, and so on)
- Configured port link type
- Operational port link type

If you have a VLAN that spans multiple STPDs, use the `show vlan <vlan_name> stpd` command to display the STP configuration of the ports assigned to that specific VLAN.

The command displays the following:

- STPD port configuration
- STPD port mode of operation
- STPD path cost
- STPD priority
- STPD state (root bridge, and so on)
- Port role (root bridge, edge port, etc.)
- STPD port state (forwarding, blocking, and so on)
- Configured port link type
- Operational port link type

11 Virtual Router Redundancy Protocol

This chapter covers the following topics:

- Overview on page 165
- Determining the VRRP Master on page 166
- Additional VRRP Highlights on page 168
- VRRP Operation on page 169
- VRRP Configuration Parameters on page 171
- VRRP Examples on page 172

This chapter assumes that you are already familiar with the Virtual Router Redundancy Protocol (VRRP). If not, refer to the following publications for additional information:

- RFC 2338—*Virtual Router Redundancy Protocol (VRRP)*
- RFC 2787—*Definitions of Managed Objects for the Virtual Router Redundancy Protocol*
- *Draft IETF VRRP Specification v2.06*

Overview

VRRP is a protocol that allows multiple switches to provide redundant routing services to users. VRRP is used to eliminate the single point of failure associated with manually configuring a default gateway address on each host in a network. Without using VRRP, if the configured default gateway fails, you must reconfigure each host on the network to use a different router as the default gateway. VRRP provides a redundant path for the hosts. If the default gateway fails, the backup router assumes forwarding responsibilities.

NOTE

IGMP snooping must be enabled for VRRP to operate correctly.

VRRP Terms

Table 33 describes terms associated with VRRP.

Table 33: VRRP Terms

Term	Description
virtual router	A VRRP router is a group of one or more physical devices that acts as the default gateway for hosts on the network. The virtual router is identified by a virtual router identifier (VRID) and an IP address.
VRRP router	Any router that is running VRRP. A VRRP router can participate in one or more virtual routers. A VRRP router can be a backup router for one more master routers.
IP address owner	A single VRRP router that has the IP address of the virtual router configured as its real interface address. The IP address owner responds to TCP/IP packets addressed to the virtual router IP address. The IP address owner is optional in a VRRP configuration.
master router	The physical device (router) in the virtual router that is responsible for forwarding packets sent to the virtual router, and responding to ARP requests. The master router sends out periodic advertisements that let backup routers on the network know that it is alive. If the IP address owner is identified, it always becomes the master.
backup router	Any VRRP router in the virtual router that is not elected as the master. The backup router is available to assume forwarding responsibility if the master becomes unavailable.
VRID	Virtual router identifier. Each virtual router is given a unique VRID. All of the VRRP routers that participate in the virtual router are assigned the same VRID.
virtual router MAC address	RFC 2338 assigns a static MAC address for the first 5 octets of the virtual router. These octets are set to 00-00-5E-00-01. When you configure the VRID, the last octet of the MAC address is dynamically assigned the VRID number.

Determining the VRRP Master

The VRRP master is determined by the following factors:

- **VRRP priority**—This is a user-defined field. The range of the priority value is 1 to 254; a higher number has higher priority. The value of 255 is reserved for a router that is configured with the virtual router IP address. A value of 0 is reserved for the master router, to indicate it is releasing responsibility for the virtual router. The default value is 100.
- **Higher IP address**—If the routers have the same configured priority, the router with the higher IP address becomes the master.

VRRP Tracking

Tracking information is used to track various forms of connectivity from the VRRP router to the outside world. ExtremeWare XOS supports the use of VRRP route table tracking.

You can configure VRRP to track specified routes in the route table as criteria for failover. If any of the configured routes are not available within the route table, the router automatically relinquishes master status and remains in backup mode.

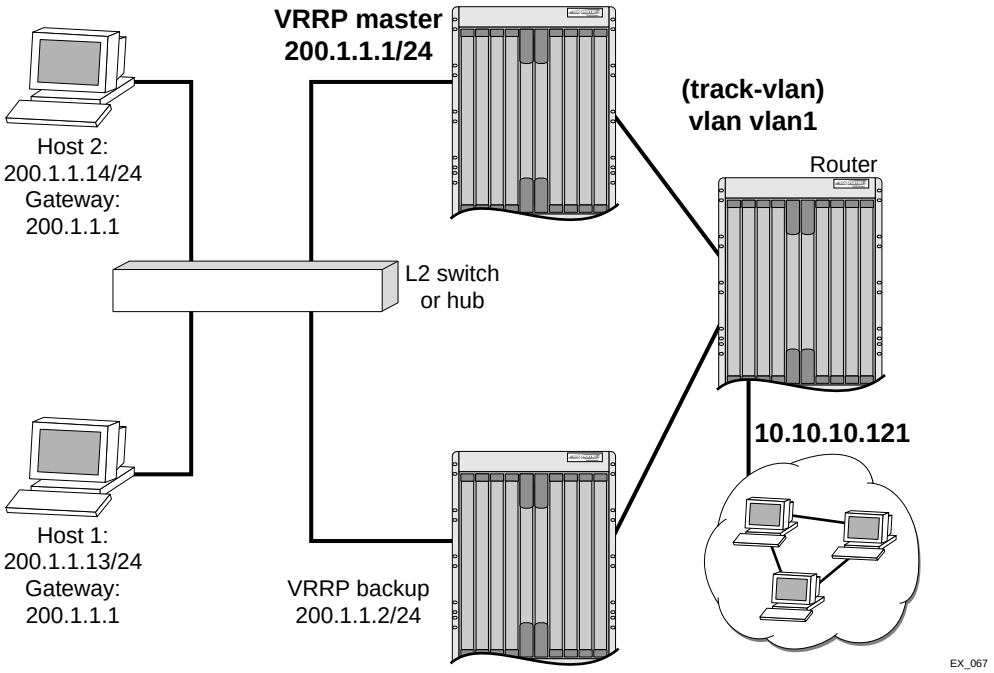
To add or delete a tracked route, use the following command:

```
config vrrp vlan <vlan_name> vrid <vridval> [add | delete] track-iproute
<ipaddress>/<masklength>
```

VRRP Tracking Example

Figure 26 is an example of VRRP tracking.

Figure 26: VRRP tracking



To configure VLAN tracking, as shown in Figure 26, use the following command:

```
Configure vlan vrrp1 add track-vlan vlan1
```

Using the tracking mechanism, if VLAN1 fails, the VRRP master realizes that there is no path to upstream router via the Master switch and implements a failover to the backup.

To configure route table tracking, as shown in Figure 26, use the following command:

```
configure vlan vrrp1 add track-iproute 10.10.10.0/24
```

The route specified in this command must exist in the IP routing table. When the route is no longer available, the switch implements a failover to the backup.

To configure ping tracking, as shown in Figure 26, use the following command:

```
configure vlan vrrp1 add track-ping 10.10.10.121 2 2
```

The specified IP address is tracked. If the fail rate is exceeded the switch implements a failover to the backup.

Electing the Master Router

VRRP uses an election algorithm to dynamically assign responsibility for the master router to one of the VRRP routers on the network. A VRRP router is elected master if the router has the highest priority (the range is 1 - 255).

If the master router becomes unavailable, the election process provides dynamic failover and the backup router that has the highest priority assumes the role of master.

A new master is elected when one of the following things happen:

- VRRP is disabled on the master router.
- Loss of communication occurs between master and backup router(s).
- Another VRRP router is attached to the VLAN, and the new router has the same priority as the current master.

When VRRP is disabled on the master interface, the master router sends an advertisement with the priority set to 0 to all backup routers. This signals the backup routers that they do not need to wait for the master down interval to expire, and the master election process for a new master can begin immediately.

The master down interval is set as follows:

3 * advertisement interval + skew time

Where:

- The advertisement interval is a user-configurable option.
- The skew time is (256-priority / 256).

NOTE

An extremely busy CPU can create a short dual master situation. To avoid this, increase the advertisement interval.

Additional VRRP Highlights

The following additional points pertain to VRRP:

- VRRP packets are encapsulated IP packets.
- The VRRP multicast address is 224.0.0.18.
- The virtual router MAC address is `00 00 5E 00 01 <vrid>`
- Duplicate virtual router IDs are allowed on the router, but not on the same interface.
- The maximum number of supported VRIDs per interface is 7.
- An interconnect link between VRRP routers should not be used, except when VRRP routers have hosts directly attached.
- A maximum of 64 VRID instances are supported on the router.
- Up to 7 unique VRIDs can be configured on the router. VRIDs can be re-used, but not on the same interface.

- VRRP and Spanning Tree can be simultaneously enabled on the same switch.

VRRP Operation

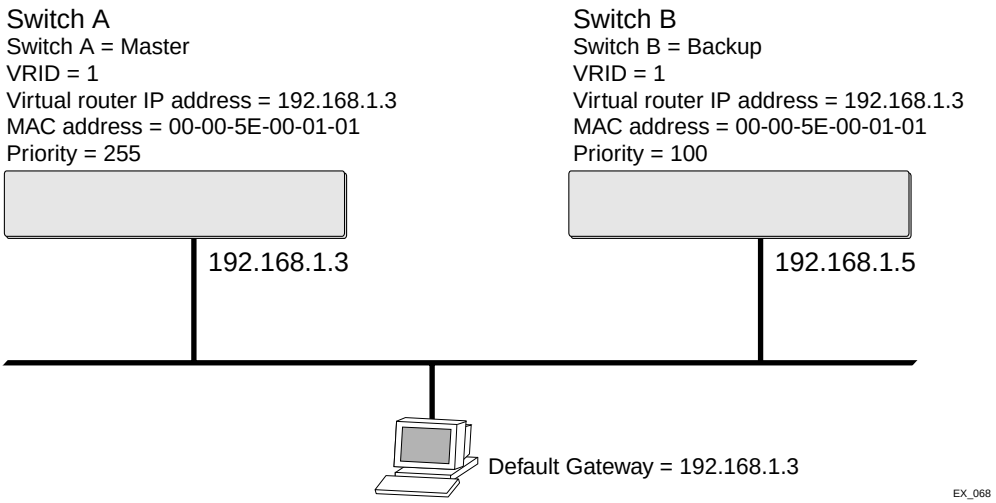
This section describes two VRRP network configuration:

- A simple VRRP network
- A fully-redundant VRRP network

Simple VRRP Network Configuration

Figure 27 shows a simple VRRP network.

Figure 27: Simple VRRP network



In Figure 27, a virtual router is configured on Switch A and Switch B using these parameters:

- VRID is 1.
- MAC address is 00-00-5E-00-01-01.
- IP address is 192.168.1.3.

Switch A is configured with a priority of 255. This priority indicates that it is the master router. Switch B is configured with a priority of 100. This indicates that it is a backup router.

The master router is responsible for forwarding packets sent to the virtual router. When the VRRP network becomes active, the master router broadcasts an ARP request that contains the virtual router MAC address (in this case, 00-00-5E-00-01-01) for each IP address associated with the virtual router. Hosts on the network use the virtual router MAC address when they send traffic to the default gateway.

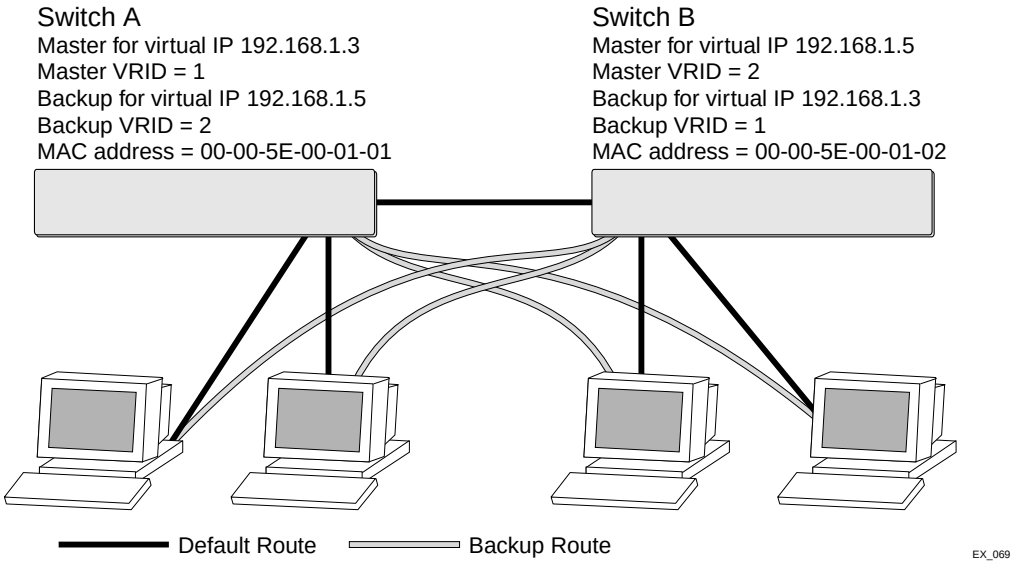
The virtual router IP address is configured to be the real interface address of the IP address owner. The IP address owner is usually the master router. The virtual router IP address is also configured on each backup router. However, in the case of the backup router, this IP address is not associated with a physical interface. Each physical interface on each backup router must have a unique IP address. The virtual router IP address is also used as the default gateway address for each host on the network.

If the master router fails, the backup router assumes forwarding responsibility for traffic addressed to the virtual router MAC address. However, because the IP address associated with the master router is not physically located on the backup router, the backup router cannot reply to TCP/IP messages (such as pings) sent to the virtual router.

Fully-Redundant VRRP Network

You can use two or more VRRP-enabled switches to provide a fully-redundant VRRP configuration on your network. Figure 28 shows a fully-redundant VRRP configuration.

Figure 28: Fully-redundant VRRP configuration



In Figure 28, switch A is configured as follows:

- IP address 192.168.1.3
- Master router for VRID 1
- Backup router for VRID 2
- MAC address 00-00-5E-00-01-01

Switch B is configured as follows:

- IP address 192.168.1.5
- Master router for VRID 2
- Backup router for VRID 1
- MAC address 00-00-5E-00-01-02

Both virtual routers are simultaneously operational. The traffic load from the four hosts is split between them. Host 1 and host 2 are configured to use VRID 1 on switch A as their default gateway. Host 3 and host 4 are configured to use VRID 2 on switch B as their default gateway. In the event that either switch fails, the backup router configured is standing by to resume normal operation.

VRRP Configuration Parameters

Table 34 lists the parameters that are configured on a VRRP router.

Table 34: VRRP Configuration Parameters

Parameter	Description
vrid	Virtual router identifier. Configured item in the range of 1- 255. This parameter has no default value.
priority	Priority value to be used by this VRRP router in the master election process. A value of 255 is reserved for a router that is configured with the virtual router IP address. A value of 0 is reserved for the master router to indicate it is releasing responsibility for the virtual router. The range is 1 - 254. The default value is 100.
ip_address	One or more IP addresses associated with this virtual router. This parameter has no default value.
advertisement_interval	Time interval between advertisements, in seconds. The range is 1 - 255. The default value is 1 second.
skew_time	Time to skew master_down_interval, in seconds. This value is calculated as ((256-priority)/256).
master_down_interval	Time interval for backup router to declare master down, in seconds. This value is calculated as ((3 * advertisement_interval) + skew_time).
preempt_mode	Controls whether a higher priority backup router preempts a lower priority master. A value of true allows preemption. A value of false prohibits preemption. The default setting is true. **NOTE** *The router that owns the virtual router IP address always preempts, independent of the setting of this parameter.*

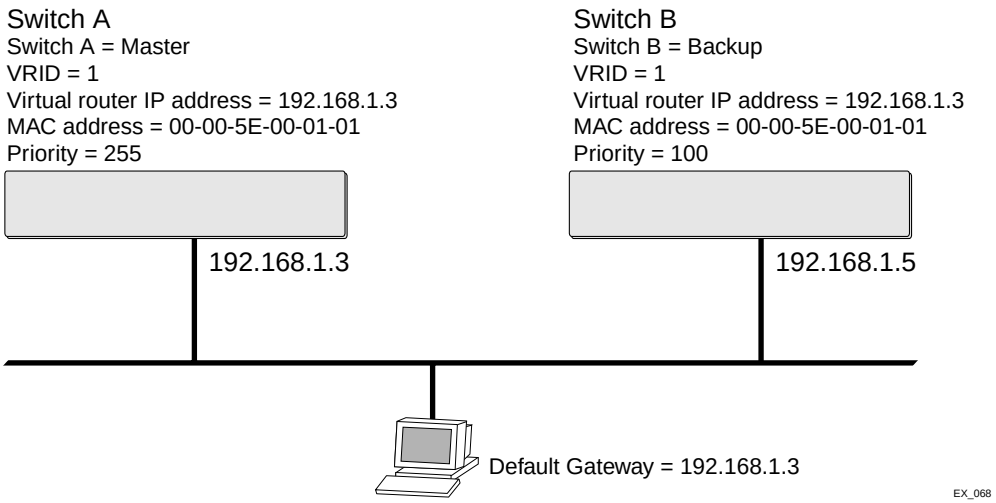
VRRP Examples

This section provides the configuration syntax for the two VRRP networks discussed in this chapter.

Configuring the Simple VRRP Network

The following illustration shows the simple VRRP network described in Figure 27.

Figure 29: Simple VRRP network



The configuration commands for switch A are as follows:

```
configure vlan vlan1 ipaddress 192.168.1.3/24
configure vrrp vlan vlan1 vrid 1
configure vrrp vlan vlan1 vrid 1 prioirty 255
configure vrrp vlan vlan1 vrid 1 add 192.168.1.3
enable vrrp
```

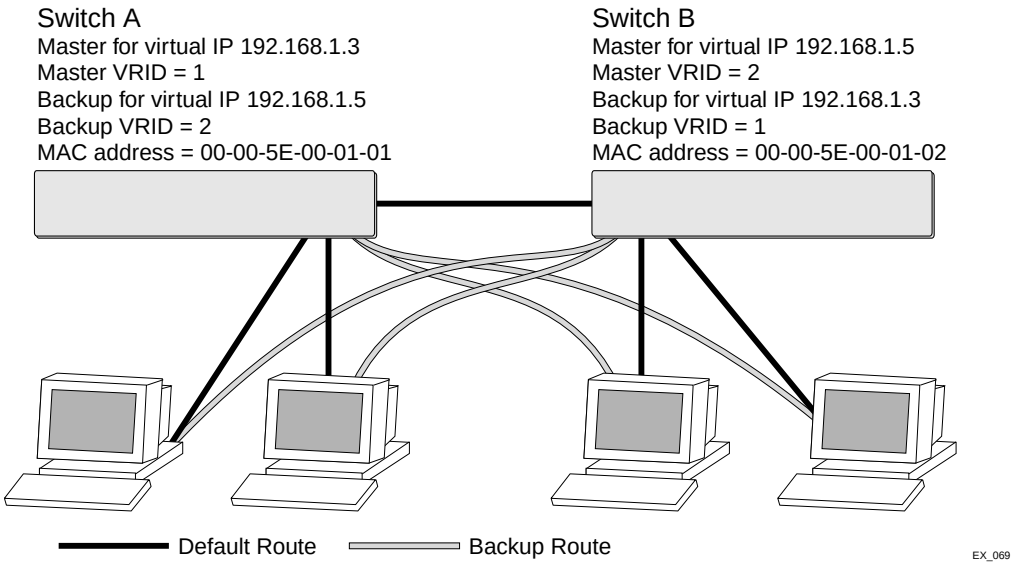
The configuration commands for switch B are as follows:

```
configure vlan vlan1 ipaddress 192.168.1.5/24
configure vrrp vlan vlan1 vrid 1
configure vrrp vlan vlan1 vrid 1 add 192.168.1.3
enable vrrp
```

Configuring the Fully-Redundant VRRP Network

The following illustration shows the fully-redundant VRRP network configuration described in Figure 28.

Figure 30: Fully-redundant VRRP configuration



The configuration commands for switch A are as follows:

```
configure vlan vlan1 ipaddress 192.168.1.3/24
create vlan vlan1 vrid 1
configure vrrp vlan vlan1 vrid 1 priority 255
configure vrrp vlan vlan1 vrid 1 add 192.168.1.3
create vlan vlan1 vrid 2
configure vrrp vlan vlan1 vrid 2 add 192.168.1.5
enable vrrp
```

The configuration commands for switch B are as follows:

```
configure vlan vlan1 ipaddress 192.168.1.5/24
create vlan vlan1 vrid 2
configure vrrp vlan vlan1 vrid 2 priority 255
configure vrrp vlan vlan1 vrid 2 add 192.168.1.5
create vlan vlan1 vrid 1
configure vrrp vlan vlan1 vrid 1 add 192.168.1.3
enable vrrp
```

12 IP Unicast Routing

This chapter describes the following topics:

- Overview of IP Unicast Routing on page 175
- Proxy ARP on page 178
- Relative Route Priorities on page 179
- Configuring IP Unicast Routing on page 179
- Routing Configuration Example on page 180
- Configuring DHCP/BOOTP Relay on page 182

This chapter assumes that you are already familiar with IP unicast routing. If not, refer to the following publications for additional information:

- RFC 1256—*ICMP Router Discovery Messages*
- RFC 1812—*Requirements for IP Version 4 Routers*

NOTE

For more information on interior gateway protocols, see Chapter 13. For information on exterior gateway protocols, see Chapter 14.

Overview of IP Unicast Routing

The switch provides full layer 3, IP unicast routing. It exchanges routing information with other routers on the network using either the Routing Information Protocol (RIP) or the Open Shortest Path First (OSPF) protocol. The switch dynamically builds and maintains a routing table, and determines the best path for each of its routes.

Each host using the IP unicast routing functionality of the switch must have a unique IP address assigned. In addition, the default gateway assigned to the host must be the IP address of the router interface.

Router Interfaces

The routing software and hardware routes IP traffic between router interfaces. A router interface is simply a VLAN that has an IP address assigned to it.

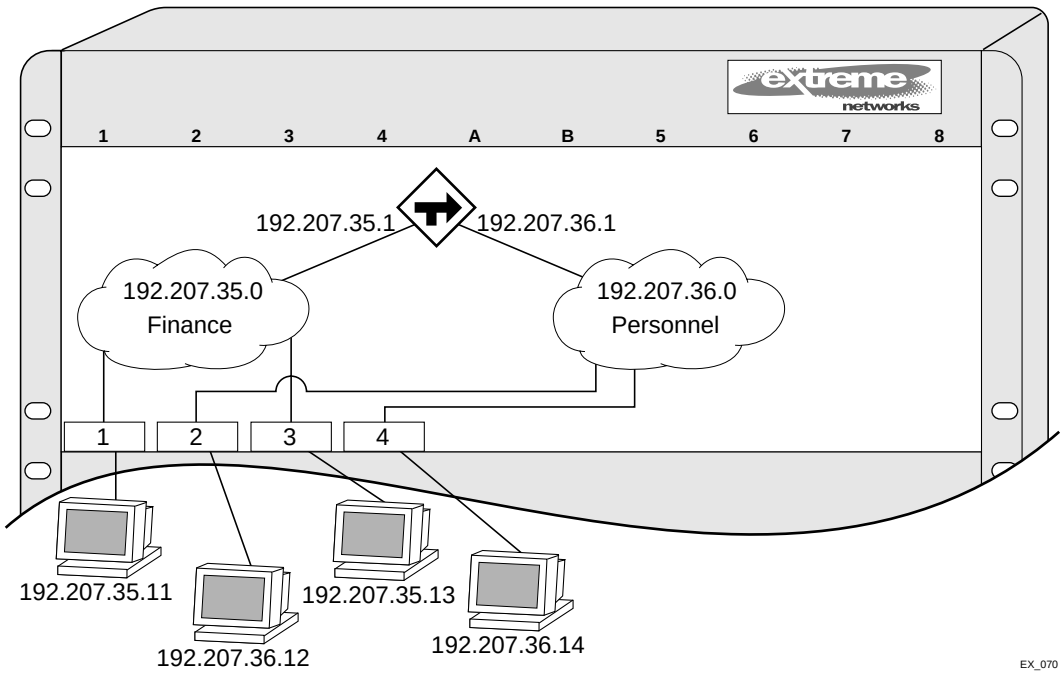
As you create VLANs with IP addresses belonging to different IP subnets, you can also choose to route between the VLANs. Both the VLAN switching and IP routing function occur within the switch.

NOTE

Each IP address and mask assigned to a VLAN must represent a unique IP subnet. You cannot configure the same IP address and subnet on different VLANs.

In Figure 31, a BlackDiamond switch is depicted with two VLANs defined; *Finance* and *Personnel*. All ports on slots 1 and 3 are assigned to *Finance*; all ports on slots 2 and 4 are assigned to *Personnel*. *Finance* belongs to the IP network 192.207.35.0; the router interface for *Finance* is assigned the IP address 192.206.35.1. *Personnel* belongs to the IP network 192.207.36.0; its router interface is assigned IP address 192.207.36.1. Traffic within each VLAN is switched using the Ethernet MAC addresses. Traffic between the two VLANs is routed using the IP addresses.

Figure 31: Routing between VLANs



Populating the Routing Table

The switch maintains an IP routing table for both network routes and host routes. The table is populated from the following sources:

- Dynamically, by way of routing protocol packets or by ICMP redirects exchanged with other routers
- Statically, by way of routes entered by the administrator
  - Default routes, configured by the administrator
  - Locally, by way of interface addresses assigned to the system
  - By other static routes, as configured by the administrator

NOTE

If you define a default route, and subsequently delete the VLAN on the subnet associated with the default route, the invalid default route entry remains. You must manually delete the configured default route.

Dynamic Routes

Dynamic routes are typically learned by way of RIP or OSPF. Routers that use RIP or OSPF exchange information in their routing tables in the form of advertisements. Using dynamic routes, the routing table contains only networks that are reachable.

Dynamic routes are aged out of the table when an update for the network is not received for a period of time, as determined by the routing protocol.

Static Routes

Static routes are manually entered into the routing table. Static routes are used to reach networks not advertised by routers.

Static routes can also be used for security reasons, to control which routes you want advertised by the router. You can decide if you want all static routes to be advertised, using one of the following commands:

- `enable rip export [bgp | direct | e-bgp | i-bgp | ospf | ospf-extern1 | ospf-extern2 | ospf-inter | ospf-intra | static] [cost <number> {tag <number>} | policy <policy-name>]` or `disable rip export [bgp | direct | e-bgp | i-bgp | ospf | ospf-extern1 | ospf-extern2 | ospf-inter | ospf-intra | static]`
- `enable ospf export [bgp | direct | e-bgp | i-bgp | rip | static] [cost <cost> type [ase-type-1 | ase-type-2] {tag <number>} | <policy-map>]` or `disable ospf export [bgp | direct | e-bgp | i-bgp | rip | static]`

The default setting is disabled. Static routes are never aged out of the routing table.

A static route must be associated with a valid IP subnet. An IP subnet is associated with a single VLAN by its IP address and subnet mask. If the VLAN is subsequently deleted, the static route entries using that subnet must be deleted manually.

Multiple Routes

When there are multiple, conflicting choices of a route to a particular destination, the router picks the route with the longest matching network mask. If these are still equal, the router picks the route using the following criteria (in the order specified):

- Directly attached network interfaces
- ICMP redirects
- Static routes
- Directly attached network interfaces that are not active.

NOTE

If you define multiple default routes, the route that has the lowest metric is used. If multiple default routes have the same lowest metric, the system picks one of the routes.

You can also configure *blackhole* routes—traffic to these destinations is silently dropped.

Proxy ARP

Proxy Address Resolution Protocol (ARP) was first invented so that ARP-capable devices could respond to ARP Request packets on behalf of ARP-incapable devices. Proxy ARP can also be used to achieve router redundancy and simplify IP client configuration. The switch supports proxy ARP for this type of network configuration. The section describes some example of how to use proxy ARP with the switch.

ARP-Incapable Devices

To configure the switch to respond to ARP Requests on behalf of devices that are incapable of doing so, you must configure the IP address and MAC address of the ARP-incapable device using the use the following command:

```
configure iparp add proxy <ip_addr> {vr <vr_name>} {<mask>} {<mac>} {always}
```

After it is configured, the system responds to ARP Requests on behalf of the device as long as the following conditions are satisfied:

- The valid IP ARP Request is received on a router interface.
- The target IP address matches the IP address configured in the proxy ARP table.
- The proxy ARP table entry indicates that the system should always answer this ARP Request, regardless of the ingress VLAN (the `always` parameter must be applied).

When all the proxy ARP conditions are met, the switch formulates an ARP Response using the configured MAC address in the packet.

Proxy ARP Between Subnets

In some networks, it is desirable to configure the IP host with a wider subnet than the actual subnet mask of the segment. Proxy ARP can be used so that the router answers ARP Requests for devices outside of the subnet. As a result, the host communicates as if all devices are local. In reality, communication with devices outside of the subnet are proxied by the router.

For example, an IP host is configured with a class B address of 100.101.102.103 and a mask of 255.255.0.0. The switch is configured with the IP address 100.101.102.1 and a mask of 255.255.255.0. The switch is also configured with a proxy ARP entry of IP address 100.101.0.0 and mask 255.255.0.0, *without* the `always` parameter.

When the IP host tries to communicate with the host at address 100.101.45.67, the IP hosts communicates as if the two hosts are on the same subnet, and sends out an IP ARP Request. The switch answers on behalf of the device at address 100.101.45.67, using its own MAC address. All subsequent data packets from 100.101.102.103 are sent to the switch, and the switch routes the packets to 100.101.45.67.

Relative Route Priorities

Table 35 lists the relative priorities assigned to routes depending upon the learned source of the route.

NOTE

Although these priorities can be changed, do not attempt any manipulation unless you are expertly familiar with the possible consequences.

Table 35: Relative Route Priorities

Route Origin	Priority
Direct	10
BlackHole	50
Static	1100
ICMP	1200
OSPFIntra	2200
OSPFInter	2300
RIP	2400
OSPFExtern1	3200
OSPFExtern2	3300
BOOTP	5000

To change the relative route priority, use the following command:

```
configure iproute priority [rip | blackhole | direct | bootp | icmp | static |
ospf-intra | ospf-inter | ospf-as-external | ospf-extern1 | ospf-extern2] <priority>
```

Configuring IP Unicast Routing

This section describes the commands associated with configuring IP unicast routing on the switch. To configure routing:

1. Create and configure two or more VLANs.
2. Assign each VLAN that will be using routing an IP address using the following command:

```
configure vlan <vlan_name> ipaddress <ipaddress> {<ipNetmask>}
```

Ensure that each VLAN has a unique IP address.

3 Configure a default route using the following command:

```
configure iproute add default <gateway> {vr <vrname>} {<metric>} {multicast-only |
unicast-only}
```

Default routes are used when the router has no other dynamic or static route to the requested destination.

4 Turn on IP routing for one or all VLANs using the following command:

```
enable ipforwarding {[vr <name> | {broadcast} {fast-direct-broadcast}
{ignore-broadcast} {vlan <name>}]}
```

5 Turn on RIP or OSPF using one of the following commands:

```
enable ripp
```

```
enable ospf
```

Verifying the IP Unicast Routing Configuration

Use the `show iproute` command to display the current configuration of IP unchaste routing for the switch, and for each VLAN. The `show iproute` command displays the currently configured routes, and includes how each route was learned.

Additional verification commands include:

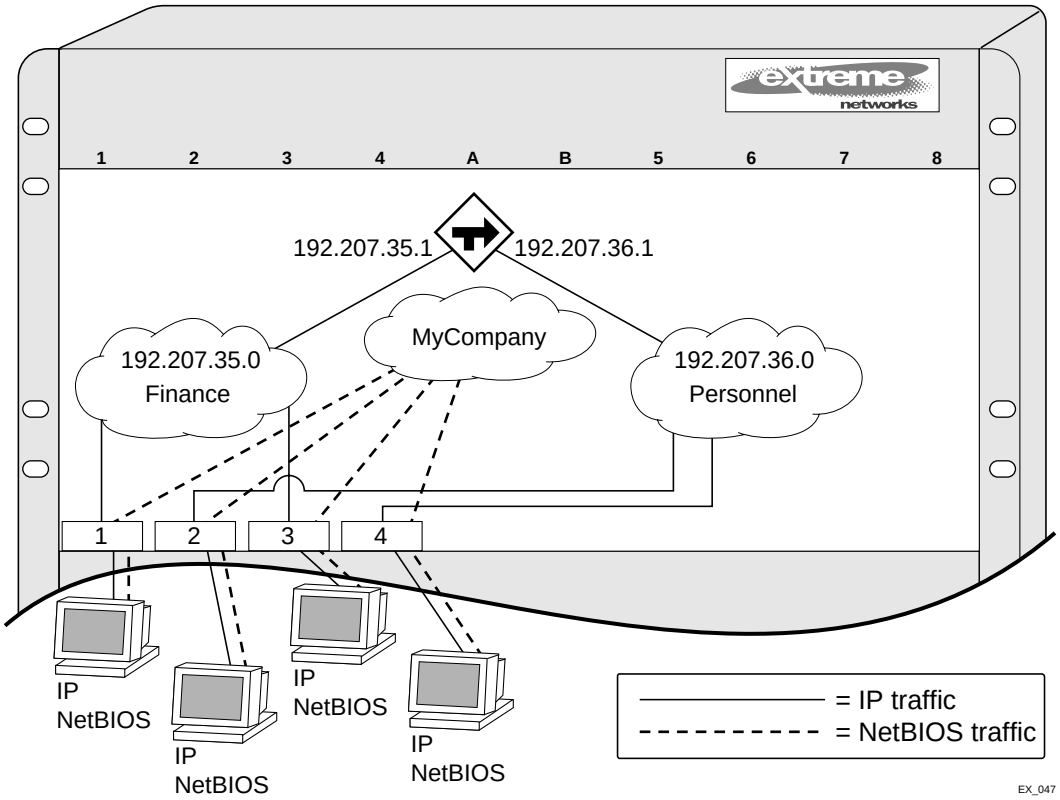
- `show bootprelay`—Displays the IP ARP table of the system.
- `show ipconfig`—Displays configuration information for one or more VLANs.

Routing Configuration Example

Figure 32 illustrates a BlackDiamond switch that has three VLANs defined as follows:

- *Finance*
  - All ports on slots 1 and 3 have been assigned.
  - IP address 192.207.35.1.
- *Personnel*
  - Protocol-sensitive VLAN using the IP protocol.
  - All ports on slots 2 and 4 have been assigned.
  - IP address 192.207.36.1.
- *MyCompany*
  - Port-based VLAN.
  - All ports on slots 1 through 4 have been assigned.

Figure 32: Unicast routing configuration example



The stations connected to the system generate a combination of IP traffic and NetBIOS traffic. The IP traffic is filtered by the protocol-sensitive VLANs. All other traffic is directed to the VLAN *MyCompany*.

In this configuration, all IP traffic from stations connected to slots 1 and 3 have access to the router by way of the VLAN *Finance*. Ports on slots 2 and 4 reach the router by way of the VLAN *Personnel*. All other traffic (NetBIOS) is part of the VLAN *MyCompany*.

The example in Figure 32 is configured as follows:

```
create vlan Finance
create vlan Personnel
create vlan MyCompany

configure Finance protocol ip
configure Personnel protocol ip

configure Finance add port 1:*,3:*
configure Personnel add port 2:*,4:*
configure MyCompany add port all

configure Finance ipaddress 192.207.35.1
configure Personnel ipaddress 192.207.36.1

configure rip add vlan Finance
configure rip add vlan Personnel

enable ipforwarding
enable rip
```

Configuring DHCP/BOOTP Relay

After IP unicast routing has been configured, you can configure the switch to forward Dynamic Host Configuration Protocol (DHCP) or BOOTP requests coming from clients on subnets being serviced by the switch and going to hosts on different subnets. This feature can be used in various applications, including DHCP services between Windows NT servers and clients running Windows 95. To configure the relay function:

1. Configure VLANs and IP unicast routing.
2. Enable the DHCP or BOOTP relay function, using the following command:

   ```
   enable bootprelay {vrid <vrid>}
   ```

3. Configure the addresses to which DHCP or BOOTP requests should be directed, using the following command:

   ```
   configure bootprelay add <ip_address> {vrid <vrid>}
   ```

To delete an entry, use the following command:

```
configure bootprelay delete [<ip_address> | all] {vrid <vrid>}
```

Verifying the DHCP/BOOTP Relay Configuration

To verify the DHCP/BOOTP relay configuration, use the following command:

```
show ipconfig
```

This command displays the configuration of the BOOTP relay service, and the addresses that are currently configured.

UDP Echo Server

You can use UDP Echo packets to measure the transit time for data between the transmitting and receiving end.

To enable UDP echo server support, use the following command:

```
rtlookup
```

To disable UDP echo server support, use the following command:

```
enable bootp vlan
```

13 Interior Gateway Protocols

This chapter describes the following topics:

- Overview on page 186
- Overview of RIP on page 187
- Overview of OSPF on page 188
- Route Re-Distribution on page 193
- RIP Configuration Example on page 196
- Configuring OSPF on page 197
- OSPF Configuration Example on page 199
- Displaying OSPF Settings on page 200

This chapter assumes that you are already familiar with IP unicast routing. If not, refer to the following publications for additional information:

- RFC 1058—*Routing Information Protocol (RIP)*
- RFC 1723—*RIP Version 2*
- RFC 2328—*OSPF Version 2*
- RFC 1765—*OSPF Database Overflow*
- RFC 2370—*The OSPF Opaque LSA Option*
- RFC 3101—*The OSPF Not-So-Stubby Area (NSSA) Option*
- *Interconnections: Bridges and Routers*
  by Radia Perlman
  ISBN 0-201-56332-0
  Published by Addison-Wesley Publishing Company

Overview

The switch supports the use of two interior gateway protocols (IGPs); the Routing Information Protocol (RIP), and the Open Shortest Path First (OSPF) protocol.

RIP is a distance-vector protocol, based on the Bellman-Ford (or distance-vector) algorithm. The distance-vector algorithm has been in use for many years, and is widely deployed and understood.

OSPF is a link-state protocol, based on the Dijkstra link-state algorithm. OSPF is a newer Interior Gateway Protocol (IGP), and solves a number of problems associated with using RIP on today's complex networks.

NOTE

RIP and OSPF can be enabled on a single VLAN.

RIP Versus OSPF

The distinction between RIP and OSPF lies in the fundamental differences between distance-vector protocols and link-state protocols. Using a distance-vector protocol, each router creates a unique routing table from summarized information obtained from neighboring routers. Using a link-state protocol, every router maintains an identical routing table created from information obtained from all routers in the autonomous system. Each router builds a shortest path tree, using itself as the root. The link-state protocol ensures that updates sent to neighboring routers are acknowledged by the neighbors, verifying that all routers have a consistent network map.

The biggest advantage of using RIP is that it is relatively simple to understand and implement, and it has been the *de facto* routing standard for many years.

RIP has a number of limitations that can cause problems in large networks, including:

- A limit of 15 hops between the source and destination networks.
- A large amount of bandwidth taken up by periodic broadcasts of the entire routing table.
- Slow convergence.
- Routing decisions based on hop count; no concept of link costs or delay.
- Flat networks; no concept of areas or boundaries.

OSPF offers many advantages over RIP, including:

- No limitation on hop count.
- Route updates multicast only when changes occur.
- Faster convergence.
- Support for load balancing to multiple routers based on the actual cost of the link.
- Support for hierarchical topologies where the network is divided into areas.

The details of RIP and OSPF are explained later in this chapter.

Overview of RIP

RIP is an Interior Gateway Protocol (IGP) first used in computer routing in the Advanced Research Projects Agency Network (ARPAnet) as early as 1969. It is primarily intended for use in homogeneous networks of moderate size.

To determine the best path to a distant network, a router using RIP always selects the path that has the least number of hops. Each router that data must traverse is considered to be one hop.

Routing Table

The routing table in a router using RIP contains an entry for every known destination network. Each routing table entry contains the following information:

- IP address of the destination network
- Metric (hop count) to the destination network
- IP address of the next router
- Timer that tracks the amount of time since the entry was last updated

The router exchanges an update message with each neighbor every 30 seconds (default value), or if there is a change to the overall routed topology (also called *triggered updates*). If a router does not receive an update message from its neighbor within the route timeout period (180 seconds by default), the router assumes the connection between it and its neighbor is no longer available.

Split Horizon

Split horizon is a scheme for avoiding problems caused by including routes in updates sent to the router from which the route was learned. Split horizon omits routes learned from a neighbor in updates sent to that neighbor.

Poison Reverse

Like split horizon, poison reverse is a scheme for eliminating the possibility of loops in the routed topology. In this case, a router advertises a route over the same interface that supplied the route, but the route uses a hop count of 16, defining it as unreachable.

Triggered Updates

Triggered updates occur whenever a router changes the metric for a route, and it is required to send an update message immediately, even if it is not yet time for a regular update message to be sent. This will generally result in faster convergence, but may also result in more RIP-related traffic.

Route Advertisement of VLANs

VLANs that are configured with an IP address, but are configured to not route IP or are not configured to run RIP, do not have their subnets advertised by RIP. Only those VLANs that are configured with an IP address and are configured to route IP and run RIP have their subnets advertised.

RIP Version 1 Versus RIP Version 2

A new version of RIP, called RIP version 2, expands the functionality of RIP version 1 to include:

- Variable-Length Subnet Masks (VLSMs).
- Support for next-hop addresses, which allows for optimization of routes in certain environments.
- Multicasting.

  RIP version 2 packets can be multicast instead of being broadcast, reducing the load on hosts that do not support routing protocols.

NOTE

If you are using RIP with supernetting/Classless Inter-Domain Routing (CIDR), you must use RIPv2 only.

Overview of OSPF

OSPF is a link-state protocol that distributes routing information between routers belonging to a single IP domain, also known as an *autonomous system* (AS). In a link-state routing protocol, each router maintains a database describing the topology of the autonomous system. Each participating router has an identical database maintained from the perspective of that router.

From the link-state database (LSDB), each router constructs a tree of shortest paths, using itself as the root. The shortest path tree provides the route to each destination in the autonomous system. When several equal-cost routes to a destination exist, traffic can be distributed among them. The cost of a route is described by a single metric.

Link-State Database

Upon initialization, each router transmits a link-state advertisement (LSA) on each of its interfaces. LSAs are collected by each router and entered into the LSDB of each router. Once all LSAs are received, the router uses the LSDB to calculate the best routes for use in the IP routing table. OSPF uses flooding to distribute LSAs between routers. Any change in routing information is sent to all of the routers in the network. All routers within an area have the exact same LSDB. Table 36 describes LSA type numbers.

Table 36: LSA Type Numbers

Type Number	Description
1	Router LSA
2	Network LSA
3	Summary LSA
4	AS summary LSA
5	AS external LSA
7	NSSA external LSA
9	Link local—Opaque
10	Area scoping—Opaque
11	AS scoping—Opaque

Database Overflow

The OSPF database overflow feature allows you to limit the size of the LSDB and to maintain a consistent LSDB across all the routers in the domain, which ensures that all routers have a consistent view of the network.

Consistency is achieved by:

- Limiting the number of external LSAs in the database of each router
- Ensuring that all routers have identical LSAs

To configure OSPF database overflow, use the following command:

```
configure ospf ase-limit <number> {timeout <seconds>}
```

where:

- `<number>`—Specifies the number of external LSAs that the system supports before it goes into overflow state. A limit value of zero disables the functionality.

  When the LSDB size limit is reached, OSPF database overflow flushes LSAs from the LSDB. OSPF database overflow flushes the same LSAs from all the routers, which maintains consistency.
- `timeout`—Specifies the timeout, in seconds, after which the system ceases to be in overflow state. A timeout value of zero leaves the system in overflow state until OSPF is disabled and re-enabled.

Opaque LSAs

Opaque LSAs are a generic OSPF mechanism used to carry auxiliary information in the OSPF database. Opaque LSAs are most commonly used to support OSPF traffic engineering.

Normally, support for opaque LSAs is auto-negotiated between OSPF neighbors. In the event that you experience interoperability problems, you can disable opaque LSAs across the entire system using the following command:

```
disable ospf capability opaque-lsa
```

To re-enable opaque LSAs across the entire system, use the following command:

```
enable ospf capability opaque-lsa
```

If your network uses opaque LSAs, we recommend that all routers on your OSPF network support opaque LSAs. Routers that do not support opaque LSAs do not store or flood them. At minimum a well-interconnected subsection of your OSPF network needs to support opaque LSAs to maintain reliability of their transmission.

On an OSPF broadcast network, the designated router (DR) must support opaque LSAs or none of the other routers on that broadcast network will reliably receive them. You can use the OSPF priority feature to give preference to an opaque-capable router, so that it becomes the elected DR.

For transmission to continue reliably across the network, the backup designated router (BDR) must also support opaque LSAs.

Areas

OSPF allows parts of a network to be grouped together into *areas*. The topology within an area is hidden from the rest of the autonomous system. Hiding this information enables a significant reduction

in LSA traffic, and reduces the computations needed to maintain the LSDB. Routing within the area is determined only by the topology of the area.

The three types of routers defined by OSPF are as follows:

- **Internal Router (IR)**—An internal router has all of its interfaces within the same area.
- **Area Border Router (ABR)**—An ABR has interfaces in multiple areas. It is responsible for exchanging summary advertisements with other ABRs.
- **Autonomous System Border Router (ASBR)**—An ASBR acts as a gateway between OSPF and other routing protocols, or other autonomous systems.

Backbone Area (Area 0.0.0.0)

Any OSPF network that contains more than one area is required to have an area configured as area 0.0.0.0, also called the *backbone*. All areas in an autonomous system must be connected to the backbone. When designing networks, you should start with area 0.0.0.0, and then expand into other areas.

NOTE

Area 0.0.0.0 exists by default and cannot be deleted or changed.

The backbone allows summary information to be exchanged between ABRs. Every ABR hears the area summaries from all other ABRs. The ABR then forms a picture of the distance to all networks outside of its area by examining the collected advertisements, and adding in the backbone distance to each advertising router.

When a VLAN is configured to run OSPF, you must configure the area for the VLAN. If you want to configure the VLAN to be part of a different OSPF area, use the following command:

```
configure ospf vlan <vlan-name> area <area-identifier>
```

If this is the first instance of the OSPF area being used, you must create the area first using the following command:

```
create ospf area <area-identifier>
```

Stub Areas

OSPF allows certain areas to be configured as *stub areas*. A stub area is connected to only one other area. The area that connects to a stub area can be the backbone area. External route information is not distributed into stub areas. Stub areas are used to reduce memory consumption and computation requirements on OSPF routers. Use the following command to configure an OSPF area as a stub area:

```
configure ospf area <area-identifier> stub [summary | nosummary] stub-default-cost
<cost>
```

Not-So-Stubby-Areas (NSSA)

NSSAs are similar to the existing OSPF stub area configuration option, but have the following two additional capabilities:

- External routes originating from an ASBR connected to the NSSA can be advertised within the NSSA.

- External routes originating from the NSSA can be propagated to other areas, including the backbone area.

The CLI command to control the NSSA function is similar to the command used for configuring a stub area, as follows:

```
configure ospf area <area-identifier> nssa [summary | nosummary] stub-default-cost
<cost> {translate}
```

The `translate` option determines whether type 7 LSAs are translated into type 5 LSAs. When configuring an OSPF area as an NSSA, the `translate` should only be used on NSSA border routers, where translation is to be enforced. If `translate` is not used on any NSSA border router in a NSSA, one of the ABRs for that NSSA is elected to perform translation (as indicated in the NSSA specification). The option should not be used on NSSA internal routers. Doing so inhibits correct operation of the election algorithm.

Normal Area

A normal area is an area that is not:

- Area 0
- Stub area
- NSSA

Virtual links can be configured through normal areas. External routes can be distributed into normal areas.

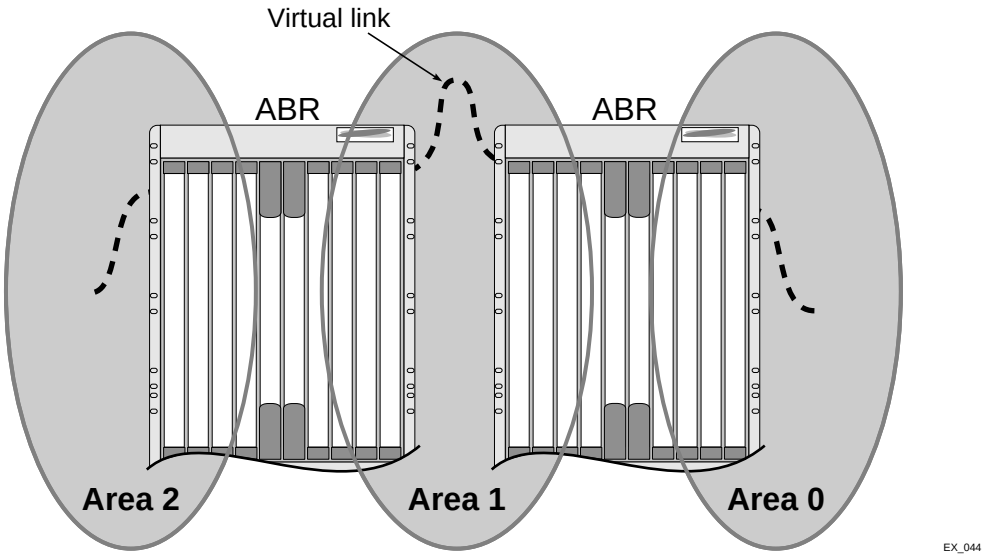
Virtual Links

In the situation when a new area is introduced that does not have a direct physical attachment to the backbone, a *virtual link* is used. A virtual link provides a logical path between the ABR of the disconnected area and the ABR of the normal area that connects to the backbone. A virtual link must be established between two ABRs that have a common area, with one ABR connected to the backbone. Figure 33 illustrates a virtual link.

NOTE

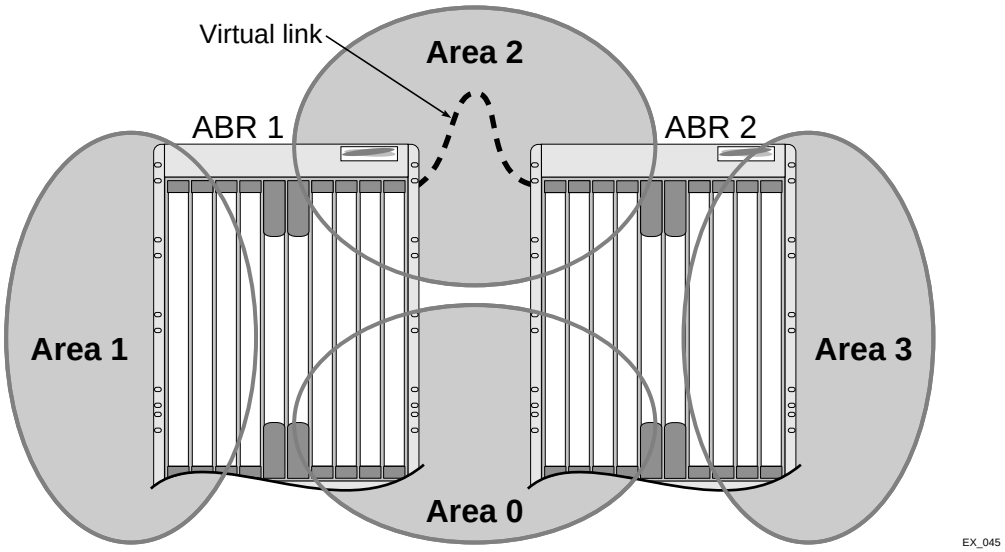
Virtual links cannot be configured through a stub or NSSA area.

Figure 33: Virtual link using Area 1 as a transit area



Virtual links are also used to repair a discontiguous backbone area. For example, in Figure 34, if the connection between ABR1 and the backbone fails, the connection using ABR2 provides redundancy so that the discontiguous area can continue to communicate with the backbone using the virtual link.

Figure 34: Virtual link providing redundancy



Point-to-Point Support

You can manually configure the OSPF link type for a VLAN. Table 37 describes the link types.

Table 37: OSPF Link Types

Link Type	Number of Routers	Description
Auto	Varies	ExtremeWare XOS automatically determines the OSPF link type based on the interface type. This is the default setting.
Broadcast	Any	Routers must elect a designated router (DR) and a backup designated router (BDR) during synchronization. Ethernet is an example of a broadcast link.
Point-to-point	Up to 2	Synchronizes faster than a broadcast link because routers do not elect a DR or BDR. Does not operate with more than two routers on the same VLAN. PPP is an example of a point-to-point link. An OSPF point-to-point link supports only zero to two OSPF routers and does not elect a DR or BDR. If you have three or more routers on the VLAN, OSPF will fail to synchronize if the neighbor is not configured.
Passive		A passive link does not send or receive OSPF packets.

NOTE

The number of routers in an OSPF point-to-point link is determined per-VLAN, not per-link.

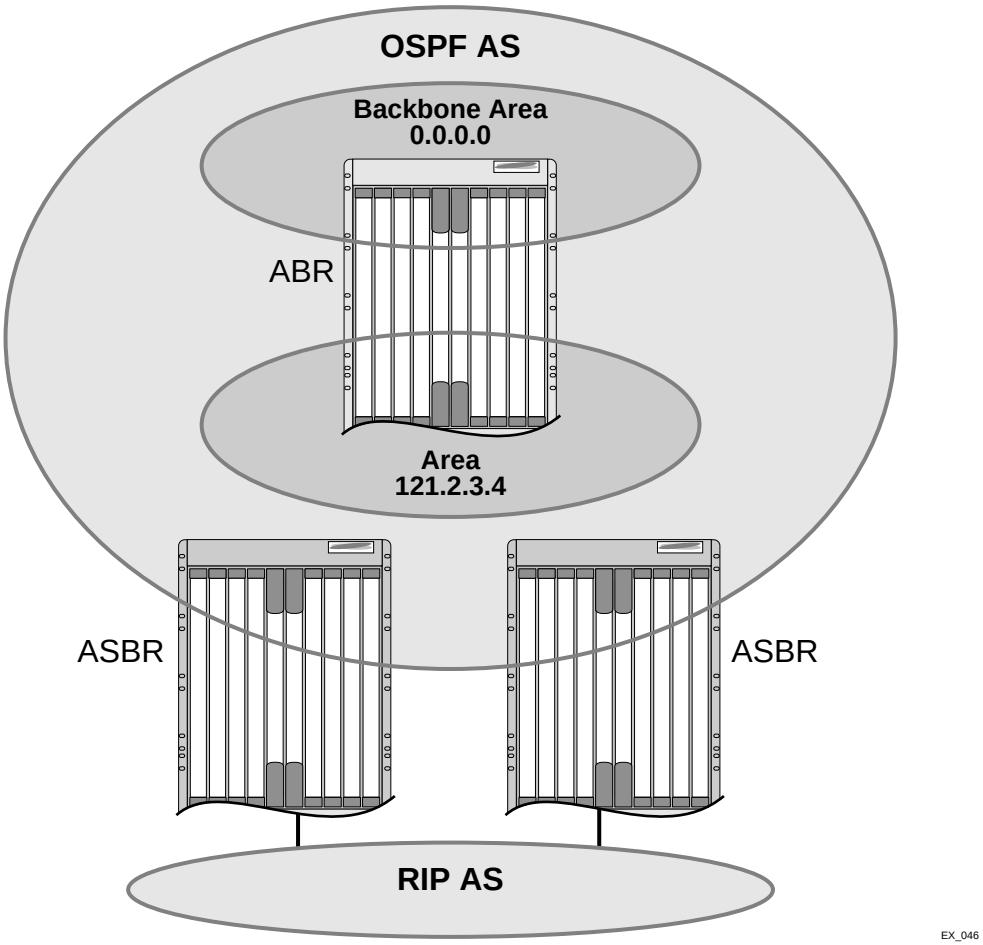
NOTE

All routers in the VLAN must have the same OSPF link type. If there is a mismatch, OSPF attempts to operate, but may not be reliable.

Route Re-Distribution

RIP and OSPF can be enabled simultaneously on the switch. Route re-distribution allows the switch to exchange routes, including static routes, between the routing protocols. Figure 35 is an example of route re-distribution between an OSPF autonomous system and a RIP autonomous system.

Figure 35: Route re-distribution



Configuring Route Re-Distribution

Exporting routes from one protocol to another, and from that protocol to the first one, are discreet configuration functions. For example, to run OSPF and RIP simultaneously, you must first configure both protocols and then verify the independent operation of each. Then you can configure the routes to export from OSPF to RIP and the routes to export from RIP to OSPF. Likewise, for any other combinations of protocols, you must separately configure each to export routes to the other.

Re-Distributing Routes into OSPF

Enable or disable the exporting of BGP, RIP, static, and direct (interface) routes to OSPF using the following commands:

```
enable ospf export [bgp | direct | e-bgp | i-bgp | rip | static] [cost <cost> type
[ase-type-1 | ase-type-2] {tag <number>} | <policy-map>]

disable ospf export [bgp | direct | e-bgp | i-bgp | rip | static]
```

These commands enable or disable the exporting of RIP, static, and direct routes by way of LSA to other OSPF routers as AS-external type 1 or type 2 routes. The default setting is disabled.

The cost metric is inserted for all BGP, RIP, static, and direct routes injected into OSPF. If the cost metric is set to 0, the cost is inserted from the route. For example, in the case of BGP export, the cost equals the MED or the path length. The tag value is used only by special routing applications. Use 0 if you do not have specific requirements for using a tag. The tag value in this instance has no relationship with 802.1Q VLAN tagging.

The same cost, type, and tag values can be inserted for all the export routes, or policies can be used for selective insertion. When a policy is associated with the export command, the policy is applied on every exported route. The exported routes can also be filtered using policies.

Verify the configuration using the command:

```
show ospf
```

Re-Distributing Routes into RIP

Enable or disable the exporting of static, direct, BGP-learned, and OSPF-learned routes into the RIP domain using the following commands:

```
enable rip export [bgp | direct | e-bgp | i-bgp | ospf | ospf-extern1 | ospf-extern2 |
ospf-inter | ospf-intra | static] [cost <number> {tag <number>} | policy
<policy-name>]

disable rip export [bgp | direct | e-bgp | i-bgp | ospf | ospf-extern1 | ospf-extern2
| ospf-inter | ospf-intra | static]
```

These commands enable or disable the exporting of static, direct, and OSPF-learned routes into the RIP domain. You can choose which types of OSPF routes are injected, or you can simply choose `ospf`, which will inject all learned OSPF routes regardless of type. The default setting is disabled.

OSPF Timers and Authentication

Configuring OSPF timers and authentication on a per-area basis is a shorthand for applying the timers and authentication to each VLAN in the area at the time of configuration. If you add more VLANs to the area, you must configure the timers and authentication for the new VLANs explicitly. Use the command:

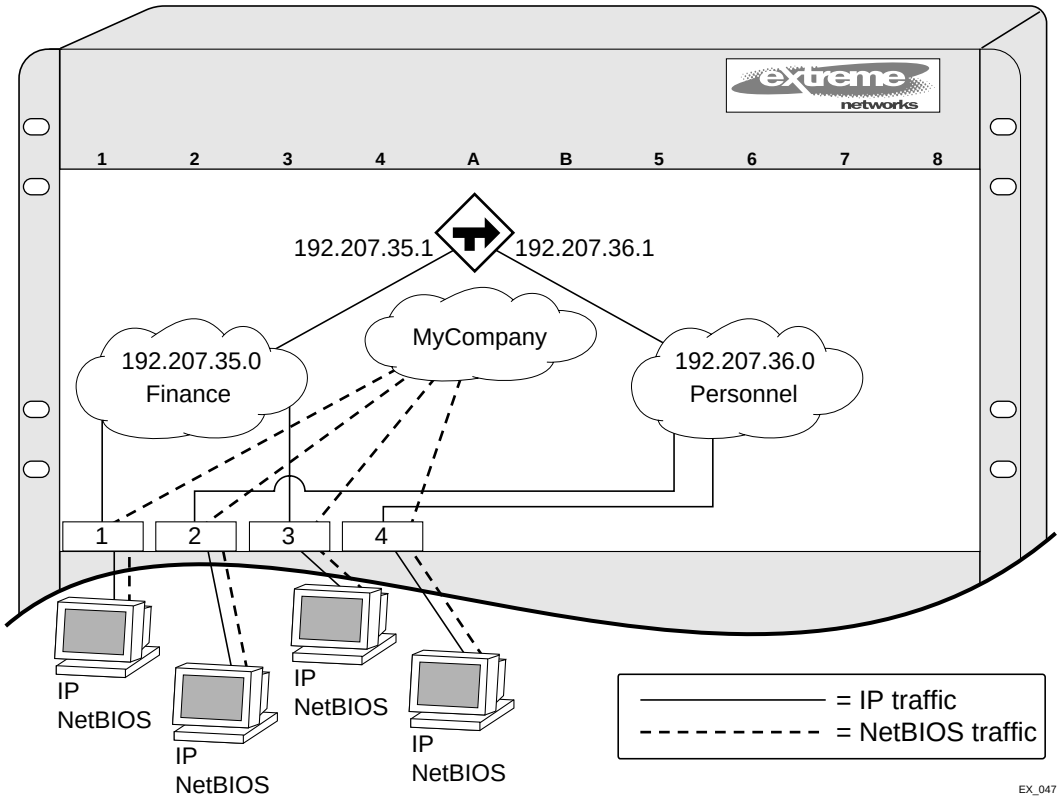
```
configure ospf vlan [<vlan-name> | all] timer <retransmit-interval>
<transit-delay> <hello-interval> <dead-interval> {<wait-timer-interval>}
```

RIP Configuration Example

Figure 36 illustrates a BlackDiamond switch that has three VLANs defined as follows:

- *Finance*
  - Protocol-sensitive VLAN using the IP protocol.
  - All ports on slots 1 and 3 have been assigned.
  - IP address 192.207.35.1.
- *Personnel*
  - Protocol-sensitive VLAN using the IP protocol.
  - All ports on slots 2 and 4 have been assigned.
  - IP address 192.207.36.1.
- *MyCompany*
  - Port-based VLAN.
  - All ports on slots 1 through 4 have been assigned.

Figure 36: RIP configuration example



The stations connected to the system generate a combination of IP traffic and NetBIOS traffic. The IP traffic is filtered by the protocol-sensitive VLANs. All other traffic is directed to the VLAN *MyCompany.*

In this configuration, all IP traffic from stations connected to slots 1 and 3 have access to the router by way of the VLAN *Finance*. Ports on slots 2 and 4 reach the router by way of the VLAN *Personnel*. All other traffic (NetBIOS) is part of the VLAN *MyCompany*.

The example in Figure 36 is configured as follows:

```
create vlan Finance
create vlan Personnel
create vlan MyCompany

configure Finance protocol ip
configure Personnel protocol ip

configure Finance add port 1:*,3:*
configure Personnel add port 2:*,4:*
configure MyCompany add port all

configure Finance ipaddress 192.207.35.1
configure Personnel ipaddress 192.207.36.1

enable ipforwarding
configure rip add vlan all
enable rip
```

Configuring OSPF

Each switch that is configured to run OSPF must have a unique router ID. It is recommended that you manually set the router ID of the switches participating in OSPF, instead of having the switch automatically choose its router ID based on the highest interface IP address. Not performing this configuration in larger, dynamic environments could result in an older link state database remaining in use.

Configuring OSPF Wait Interval

ExtremeWare XOS allows you to configure the OSPF wait interval, rather than using the router dead interval.

CAUTION

Do not configure OSPF timers unless you are comfortable exceeding OSPF specifications. Non-standard settings might not be reliable under all circumstances.

To specify the timer intervals, use the following commands:

```
configure ospf area <area-identifier> timer <retransmit-interval> <transit-delay>
<hello-interval> <dead-interval> {<wait-timer-interval>}

configure ospf virtual-link <router-identifier> <area-identifier> timer
<retransmit-interval> <transit-delay> <hello-interval> <dead-interval>
{<wait-timer-interval>}
```

```
configure ospf vlan [<vlan-name> | all] timer <retransmit-interval> <transit-delay>
<hello-interval> <dead-interval> {<wait-timer-interval>}
```

You can configure the following parameters:

- **Retransmit interval**—The length of time that the router waits before retransmitting an LSA that is not acknowledged. If you set an interval that is too short, unnecessary retransmissions will result. The default value is 5 seconds.
- **Transit delay**—The length of time it takes to transmit an LSA packet over the interface. The transit delay must be greater than 0.
- **Hello interval**—The interval at which routers send hello packets. Smaller times allow routers to discover each other more quickly, but also increase network traffic. The default value is 10 seconds.
- **Dead router wait interval (Dead Interval)**—The interval after which a neighboring router is declared down due to the fact that hello packets are no longer received from the neighbor. This interval should be a multiple of the hello interval. The default value is 40 seconds.
- **Router wait interval (Wait Timer Interval)**—The interval between the interface coming up and the election of the DR and BDR. This interval should be greater than the hello interval. If it is close to the hello interval, the network synchronizes very quickly, but might not elect the correct DR or BDR. The default value is equal to the dead router wait interval.

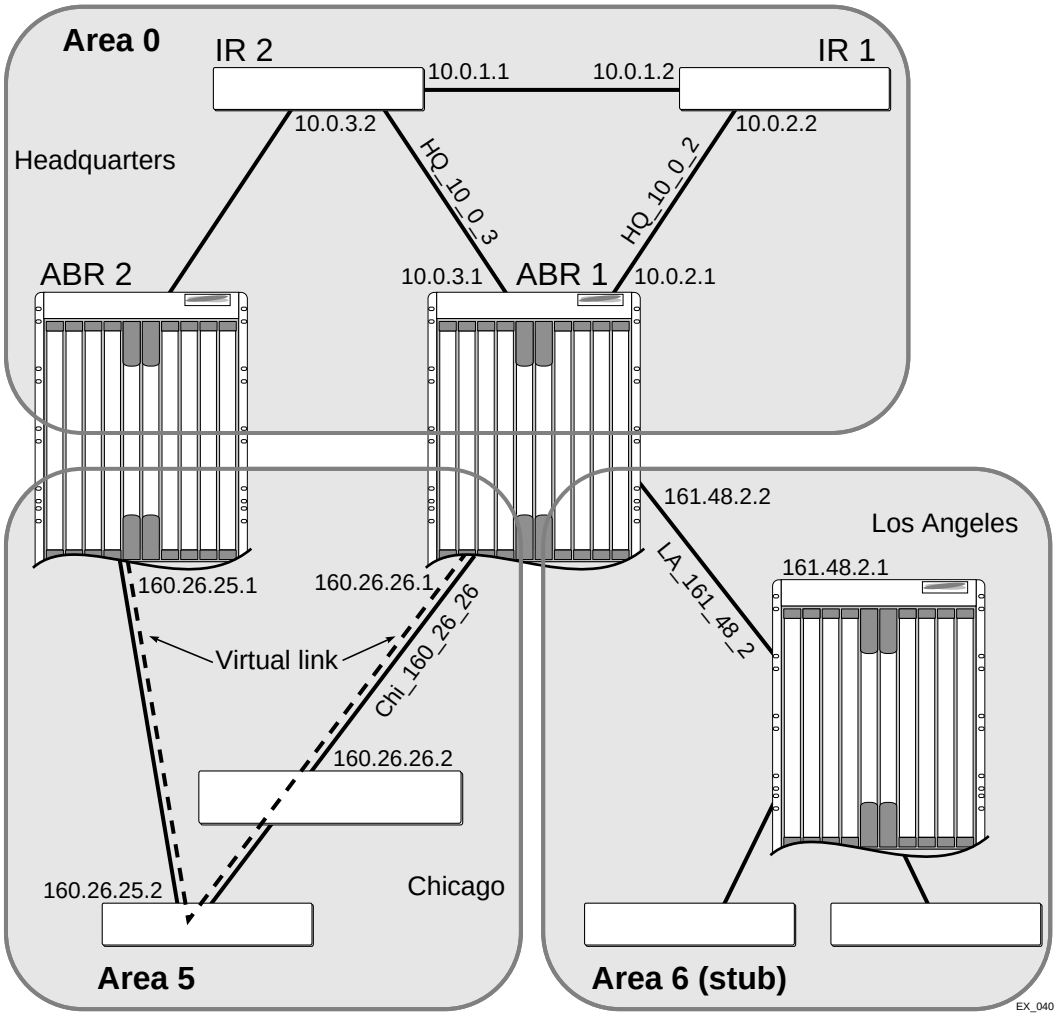
NOTE

The OSPF standard specifies that wait times are equal to the dead router wait interval.

OSPF Configuration Example

Figure 37 is an example of an autonomous system using OSPF routers. The details of this network follow.

Figure 37: OSPF configuration example



Area 0 is the backbone area. It is located at the headquarters and has the following characteristics:

- Two internal routers (IR1 and IR2)
- Two area border routers (ABR1 and ABR2)
- Network number 10.0.x.x
- Two identified VLANs (HQ_10_0_2 and HQ_10_0_3)

Area 5 is connected to the backbone area by way of ABR1 and ABR2. It is located in Chicago and has the following characteristics:

- Network number 160.26.x.x
- One identified VLAN (Chi_160_26_26)

- Two internal routers

Area 6 is a stub area connected to the backbone by way of ABR1. It is located in Los Angeles and has the following characteristics:

- Network number 161.48.x.x
- One identified VLAN (LA_161_48_2)
- Three internal routers
- Uses default routes for inter-area routing

Two router configurations for the example in Figure 37 are provided in the following section.

Configuration for ABR1

The router labeled ABR1 has the following configuration:

```
create vlan HQ_10_0_2
create vlan HQ_10_0_3
create vlan LA_161_48_2
create vlan Chi_160_26_26

configure vlan HQ_10_0_2 ipaddress 10.0.2.1 255.255.255.0
configure vlan HQ_10_0_3 ipaddress 10.0.3.1 255.255.255.0
configure vlan LA_161_48_2 ipaddress 161.48.2.2 255.255.255.0
configure vlan Chi_160_26_26 ipaddress 160.26.26.1 255.255.255.0

create ospf area 0.0.0.5
create ospf area 0.0.0.6

enable ipforwarding

configure ospf area 0.0.0.6 stub nosummary stub-default-cost 10
configure ospf add vlan LA_161_48_2 area 0.0.0.6
configure ospf add vlan Chi_160_26_26 area 0.0.0.5
configure ospf add vlan HQ_10_0_2 area 0.0.0.0
configure ospf add vlan HQ_10_0_3 area 0.0.0.0

enable ospf
```

Configuration for IR1

The router labeled IR1 has the following configuration:

```
configure vlan HQ_10_0_1 ipaddress 10.0.1.2 255.255.255.0
configure vlan HQ_10_0_2 ipaddress 10.0.2.2 255.255.255.0
enable ipforwarding
configure ospf add vlan all area 0.0.0.0
enable ospf
```

Displaying OSPF Settings

You can use a number of commands to display settings for OSPF. To show global OSPF information, use the `show ospf` command with no options.

To display information about one or all OSPF areas, use the following command:

```
show ospf area {<area-identifier>}
```

The `detail` option displays information about all OSPF areas in a detail format.

To display information about OSPF interfaces for an area, a VLAN, or for all interfaces, use the following command:

```
show ospf interfaces {vlan <vlan-name> | area <area-identifier>}
```

The `detail` option displays information about all OSPF interfaces in a detail format.

OSPF LSDB Display

ExtremeWare XOS provides several filtering criteria for the `show ospf lsdb` command. You can specify multiple search criteria and only results matching all of the criteria are displayed. This allows you to control the displayed entries in large routing tables.

To display the current link-state database, use the following command:

```
show ospf lsdb {detail | stats} {area [<area-identifier> | all]} {{lstype} [<lstype> |
all]} {lsid <lsid-address>{<lsid-mask>}} {routerid <routerid-address>
{<routerid-mask>}} {interface[[<ip-address>{<ip-mask>} | <ipNetmask>] | vlan
<vlan-name>]}
```

The `detail` option displays all fields of matching LSAs in a multi-line format. The `summary` option displays several important fields of matching LSAs, one line per LSA. The `stats` option displays the number of matching LSAs, but not any of their contents. If not specified, the default is to display in the summary format.

A common use of this command is to omit all optional parameters, resulting in the following shortened form:

```
show ospf lsdb
```

The shortened form displays all areas and all types in a summary format.

14 Exterior Gateway Routing Protocols

This chapter covers the following topics:

- Overview on page 204
- BGP Attributes on page 204
- BGP Communities on page 205
- BGP Features on page 205

This chapter describes how to configure the Border Gateway Protocol (BGP), an exterior routing protocol available on the switch.

For more information on BGP, refer to the following documents:

- RFC 1771—*Border Gateway Protocol version 4 (BGP-4)*
- RFC 1965—*Autonomous System Confederations for BGP*
- RFC 1966—*BGP Route Reflection*
- RFC 1997—*BGP Communities Attribute*
- RFC 1745—*BGP/OSPF Interaction*
- RFC 2385—*Protection of BGP Sessions via the TCP MD5 Signature Option*
- RFC 2439—*BGP Route Flap Damping*
- RFC 2796—*BGP Route Reflection - An Alternative to Full Mesh IBGP*
- RFC 2842—*Capabilities Advertisement with BGP-4*
- RFC 2858—*Multiprotocol Extensions for BGP-4*
- RFC 2918—*Route Refresh Capability for BGP-4*

NOTE

ExtremeWare XOS supports BGP version 4 only.

Overview

BGP is an exterior routing protocol that was developed for use in TCP/IP networks. The primary function of BGP is to allow different autonomous systems (ASs) to exchange network reachability information.

An autonomous system is a set of routers that are under a single technical administration. This set of routers uses a different routing protocol (such as OSPF) for intra-AS routing. One or more routers in the AS are configured to be border routers, exchanging information with other border routers (in different autonomous systems) on behalf of all of the intra-AS routers.

BGP can be used as an exterior gateway protocol (EBGP), or it can be used within an AS as an interior gateway protocol (IBGP).

BGP Attributes

The following BGP attributes are supported by the switch:

- Origin – Defines the origin of the route. Possible values are IGP, EGP, and incomplete.
- AS_Path – The list of ASs that are traversed for this route.
- Next_hop – The IP address of the next hop BGP router to reach the destination listed in the NLRI field.
- Multi_Exist_Discriminator – Used to select a particular border router in another AS when multiple border routers exist.
- Local_Preference – Used to advertise this router's degree of preference to other routers within the AS.
- Atomic_aggregate – Indicates that the sending border router has used a route aggregate prefix in the route update.
- Aggregator – Identifies the BGP router AS number and IP address that performed route aggregation.
- Community – Identifies a group of destinations that share one or more common attributes.
- Cluster_ID – Specifies a 4-byte field used by a route reflector to recognize updates from other route reflectors in the same cluster.
- Originator_ID – Specifies the router ID of the originator of the route in the local AS.
- Multiprotocol Reachable NLRI – This is an optional attribute and is used to:
  - Advertise a feasible route to a peer.
  - Permit a router to advertise the Network Layer address of the router that should be used as the next hop to the destinations listed in the Network Layer Reachability Information field of the MP_NLRI attribute.
  - Allow a given router to report some or all of the Subnetwork Points of Attachment (SNPAs) that exist within the local system.
- Multiprotocol Unreachable NLRI – This is an optional attribute that can be used for the purpose of withdrawing multiple unfeasible routes from service.

BGP Communities

A BGP community is a group of BGP destinations that require common handling. ExtremeWare XOS supports the following well-known BGP community attributes:

- no-export
- no-advertise
- no-export-subconfed

BGP Features

This section describes the following BGP features supported by ExtremeWare XOS:

- Route Reflectors on page 205
- Route Confederations on page 207
- Route Aggregation on page 211
- Using the Loopback Interface on page 211
- BGP Peer Groups on page 211
- BGP Route Flap Dampening on page 212
- BGP Route Selection on page 213
- Route Re-Distribution on page 214
- BGP Static Network on page 215

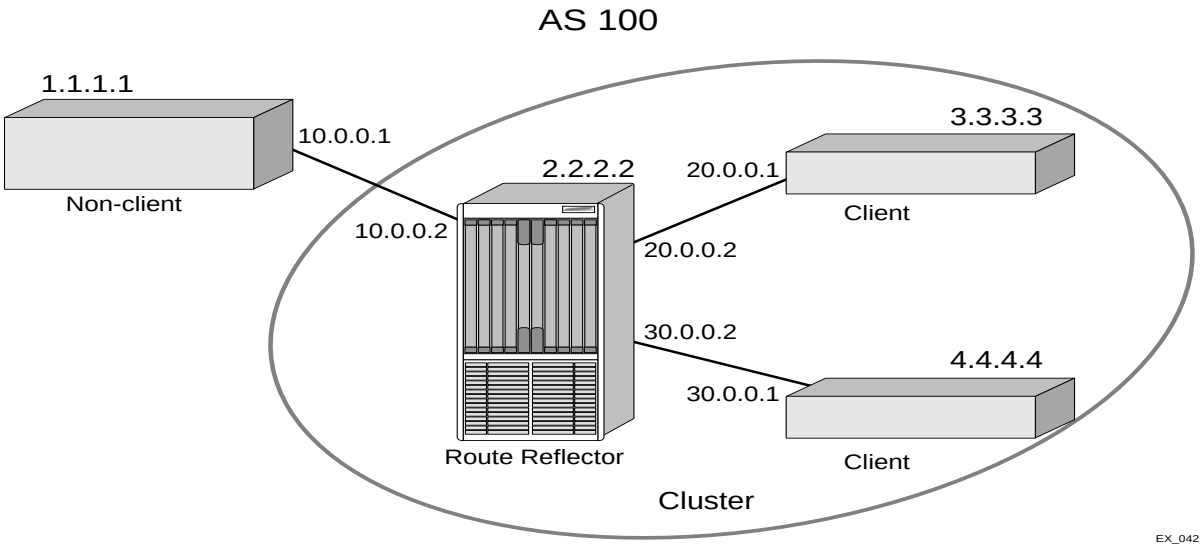
Route Reflectors

Another way to overcome the difficulties of creating a fully-meshed AS is to use *route reflectors*. Route reflectors allow a single router to serve as a central routing point for the AS or sub-AS.

A *cluster* is formed by the route reflector and its client routers. Peer routers that are not part of the cluster must be fully meshed according to the rules of BGP.

A BGP cluster, including the route reflector and its clients, is shown in Figure 38.

Figure 38: Route reflectors



The topology shown in Figure 38 minimizes the number of BGP peering sessions required in an AS by making use of route reflectors.

In this example, although the BGP speakers 3.3.3.3 and 4.4.4.4 do not have a direct BGP peering session between them, they will still receive routes from each other indirectly through 2.2.2.2. The router 2.2.2.2 is called a route reflector, and is responsible for reflecting routes between its clients. Routes received from the client 3.3.3.3 by it will be reflected to 4.4.4.4 and vice versa. Routes received from 1.1.1.1 will be reflected to all clients.

To configure router 1.1.1.1, use the following commands:

```
create vlan to_rr
config vlan to_rr add port 1:1
config vlan to_rr ipaddress 10.0.0.1/24
enable ipforwarding vlan to_rr

config bgp router 1.1.1.1
config bgp as-number 100
create bgp neighbor 10.0.0.2 remote-as 100
enable bgp
enable bgp neighbor all
```

To configure router 2.2.2.2, the route reflector, use the following commands:

```
create vlan to_nc
config vlan to_nc add port 1:1
config vlan to_nc ipaddress 10.0.0.2/24
enable ipforwarding vlan to_nc

create vlan to_c1
config vlan to_c1 add port 1:2
config vlan to_c1 ipaddress 20.0.0.2/24
```

```
enable ipforwarding vlan to_c1

create vlan to_c2
config vlan to_c2 add port 1:2
config vlan to_c2 ipaddress 30.0.0.2/24
enable ipforwarding vlan to_c2

config bgp router 2.2.2.2
config bgp as-number 100
create bgp neighbor 10.0.0.1 remote-as 100
create bgp neighbor 20.0.0.1 remote-as 100
create bgp neighbor 30.0.0.1 remote-as 100
config bgp neighbor 20.0.0.1 route-reflector-client
config bgp neighbor 30.0.0.1 route-reflector-client
enable bgp neighbor all
enable bgp
```

To configure router 3.3.3.3, use the following commands:

```
create vlan to_rr
config vlan to_rr add port 1:1
config vlan to_rr ipaddress 20.0.0.1/24
enable ipforwarding vlan to_rr

config bgp router 3.3.3.3
config bgp as-number 100
create bgp neighbor 20.0.0.2 remote-as 100
enable bgp neighbor all
enable bgp
```

To configure router 4.4.4.4, use the following commands:

```
create vlan to_rr
config vlan to_rr add port 1:1
config vlan to_rr ipaddress 30.0.0.1/24
enable ipforwarding vlan to_rr

config bgp router 4.4.4.4
config bgp as-number 100
create bgp neighbor 30.0.0.2 remote-as 100
enable bgp neighbor all
enable bgp
```

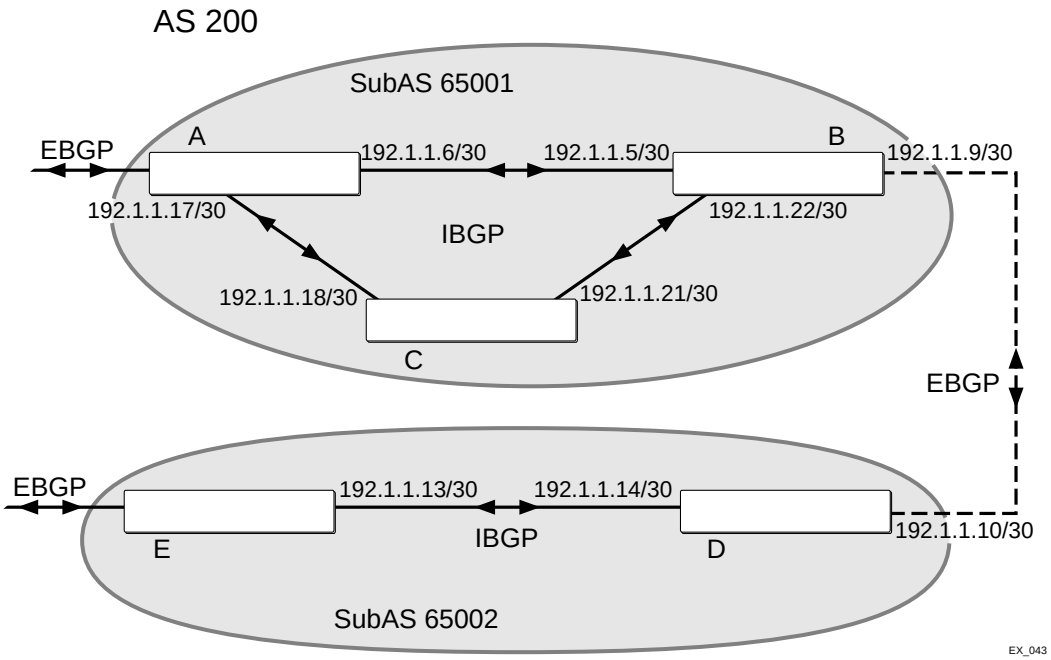
Route Confederations

BGP requires networks to use a fully-meshed router configuration. This requirement does not scale well, especially when BGP is used as an interior gateway protocol. One way to reduce the size of a fully-meshed AS is to divide the AS into multiple sub-autonomous systems and group them into a *routing confederation*. Within the confederation, each sub-AS must be fully-meshed. The confederation is advertised to other networks as a single AS.

Route Confederation Example

Figure 39 shows an example of a confederation.

Figure 39: Routing confederation



In this example, AS 200 has five BGP speakers. Without a confederation, BGP would require that the routes in AS 200 be fully meshed. Using the confederation, AS 200 is split into two sub-ASs: AS65001 and AS65002. Each sub-AS is fully meshed, and IBGP is running among its members. EBGP is used between sub-AS 65001 and sub-AS 65002. Router B and router D are EBGP peers. EBGP is also used between the confederation and outside ASs.

To configure router A, use the following commands:

```
create vlan ab
configure vlan ab add port 1
configure vlan ab ipaddress 192.1.1.6/30
enable ipforwarding vlan ab
configure ospf add vlan ab area 0.0.0.0

create vlan ac
configure vlan ac add port 2
configure vlan ac ipaddress 192.1.1.17/30
enable ipforwarding vlan ac
configure ospf add vlan ac area 0.0.0.0

disable bgp
configure bgp as-number 65001
configure bgp routerid 192.1.1.17
configure bgp confederation-id 200
enable bgp

create bgp neighbor 192.1.1.5 remote-AS-number 65001
```

```
create bgp neighbor 192.1.1.18 remote-AS-number 65001
enable bgp neighbor all
```

To configure router B, use the following commands:

```
create vlan ba
configure vlan ba add port 1
configure vlan ba ipaddress 192.1.1.5/30
enable ipforwarding vlan ba
configure ospf add vlan ba area 0.0.0.0

create vlan bc
configure vlan bc add port 2
configure vlan bc ipaddress 192.1.1.22/30
enable ipforwarding vlan bc
configure ospf add vlan bc area 0.0.0.0

create vlan bd
configure vlan bd add port  3
configure vlan bd ipaddress 192.1.1.9/30
enable ipforwarding vlan bd
configure ospf add vlan bd area 0.0.0.0

disable bgp
configure bgp as-number 65001
configure bgp routerid 192.1.1.22
configure bgp confederation-id 200
enable bgp

create bgp neighbor 192.1.1.6 remote-AS-number 65001
create bgp neighbor 192.1.1.21 remote-AS-number 65001
create bgp neighbor 192.1.1.10 remote-AS-number 65002
configure bgp add confederation-peer sub-AS-number 65002
enable bgp neighbor all
```

To configure router C, use the following commands:

```
create vlan  ca
configure vlan ca add port 1
configure vlan ca ipaddress 192.1.1.18/30
enable ipforwarding vlan ca
configure ospf add vlan ca area 0.0.0.0

create vlan cb
configure vlan cb add port 2
configure vlan cb ipaddress 192.1.1.21/30
enable ipforwarding vlan cb
configure ospf add vlan cb area 0.0.0.0

disable bgp
configure bgp as-number 65001
configure bgp routerid 192.1.1.21
configure bgp confederation-id 200
enable bgp

create bgp neighbor 192.1.1.22 remote-AS-number 65001
create bgp neighbor 192.1.1.17 remote-AS-number 65001
```

```
enable bgp neighbor all
```

To configure router D, use the following commands:

```
create vlan db
configure vlan db add port 1
configure vlan db ipaddress 192.1.1.10/30
enable ipforwarding vlan db
configure ospf add vlan db area 0.0.0.0

create vlan de
configure vlan de add port 2
configure vlan de ipaddress 192.1.1.14/30
enable ipforwarding vlan de
configure ospf add vlan de area 0.0.0.0

disable bgp
configure bgp as-number 65002
configure bgp routerid 192.1.1.14
configure bgp confederation-id 200
enable bgp

create bgp neighbor 192.1.1.9 remote-AS-number 65001
create bgp neighbor 192.1.1.13 remote-AS-number 65002
configure bgp add confederation-peer sub-AS-number 65001
enable bgp neighbor all
```

To configure router E, use the following commands:

```
create vlan ed
configure vlan ed add port  1
configure vlan ed ipaddress 192.1.1.13/30
enable ipforwarding vlan ed
configure ospf add vlan ed area 0.0.0.0

disable bgp
configure bgp as-number 65002
configure bgp routerid 192.1.1.13
configure bgp confederation-id 200
enable bgp

create bgp neighbor 192.1.1.14 remote-AS-number 65002
enable bgp neighbor 192.1.1.14
```

Route Aggregation

Route aggregation is the process of combining the characteristics of several routes so that they are advertised as a single route. Aggregation reduces the amount of information that a BGP speaker must store and exchange with other BGP speakers. Reducing the information that is stored and exchanged also reduces the size of the routing table.

Using Route Aggregation

To use BGP route aggregation:

1. Enable aggregation using the following command:

   ```
   enable bgp aggregation
   ```

2. Create an aggregate route using the following commands:

   ```
   configure bgp add aggregate-address {address-family [ipv4-unicast |
   ipv4-multicast]} <ipaddress> {as-match | as-set} {summary-only} {advertise-policy
   <policy>} {attribute-policy <policy>}
   ```

Using the Loopback Interface

If you are using BGP as your interior gateway protocol, you may decide to advertise the interface as available, regardless of the status of any particular interface. The loopback interface can also be used for EBGP multihop. Using the loopback interface eliminates multiple, unnecessary route changes.

BGP Peer Groups

You can use BGP peer groups to group together up to 512 BGP neighbors. All neighbors within the peer group inherit the parameters of the BGP peer group. The following mandatory parameters are shared by all neighbors in a peer group:

- remote AS
- source-interface
- route-policy
- send-community
- next-hop-self

Each BGP peer group is assigned a unique name when it is created. To create or delete peer groups, use the following command:

```
create bgp peer-group <peer-group-name>
delete bgp peer-group <peer-group-name>
```

Changes made to the parameters of a peer group are applied to all neighbors in the peer group. Modifying the following parameters will automatically disable and enable the neighbors before changes take effect:

- remote-as
- timer
- source-interface
- soft-in-reset

- password

Adding Neighbors to a BGP Peer Group

To create a new neighbor and add it to a BGP peer group, use the following command:

```
create bgp neighbor <remoteaddr> peer-group <peer-group-name> {multi-hop}
```

The new neighbor is created as part of the peer group and inherits all of the existing parameters of the peer group. The peer group must have remote AS configured.

To add an existing neighbor to a peer group, use the following command:

```
configure bgp neighbor [all | <remoteaddr>] peer-group [<peer-group-name> | none]
{acquire-all}
```

If you do not specify acquire-all, only the mandatory parameters are inherited from the peer group. If you specify acquire-all, all of the parameters of the peer group are inherited. This command disables the neighbor before adding it to the peer group.

To remove a neighbor from a peer group, use the `peer-group none` option.

When you remove a neighbor from a peer group, it retains the parameter settings of the group. The parameter values are not reset to those the neighbor had before it inherited the peer group values.

BGP Route Flap Dampening

Route flap dampening is a BGP feature designed to minimize the propagation of flapping routes across an internetwork. A route is considered to be flapping when it is repeatedly available, then unavailable, then available, then unavailable, and so on. When a route becomes unavailable, a Withdrawal message is sent to other connected routers, which in turn propagate the Withdrawal message to other routers. As the route becomes available again, an Advertisement message is sent and propagated throughout the network. As a route repeatedly changes from available to unavailable, large numbers of messages propagate throughout the network. This is a problem in an internetwork connected to the Internet because a route flap in the Internet backbone usually involves many routes.

Minimizing the Route Flap

The route flap dampening feature minimizes the flapping problem as follows. Suppose that the route to network 172.25.0.0 flaps. The router (in which route dampening is enabled) assigns network 172.25.0.0 a penalty of 1000 and moves it to a “history” state in which the penalty value is monitored. The router continues to advertise the status of the route to neighbors. The penalties are cumulative. When the route flaps so often that the penalty exceeds a configurable suppress limit, the router stops advertising the route to network 172.25.0.0, regardless of how many times it flaps. Thus, the route is dampened.

The penalty placed on network 172.25.0.0 is decayed until the reuse limit is reached, upon which the route is once again advertised. At half of the reuse limit, the dampening information for the route to network 172.25.0.0 is removed.

The penalty is decayed by reducing the penalty value by one-half at the end of a configurable time period, called the half-life. Routes that flap many times may reach a maximum penalty level, or ceiling, after which no additional penalty is added. The ceiling value is not directly configurable, but the configuration parameter used in practice is the maximum route suppression time. No matter how often a route has flapped, once it stops flapping, it will again be advertised after the maximum route suppression time.

Configuring Route Flap Dampening

BGP route flap dampening can be enabled on a per BGP peer session basis, for a BGP peer group, or for a set of routes, using a route map.

To enable route flap dampening over BGP peer sessions, use the following command:

```
configure bgp neighbor [all | <remoteaddr>] {address-family [ipv4-unicast |
ipv4-multicast]} dampening {{half-life <half-life-minutes> {reuse-limit
<reuse-limit-number> suppress-limit <suppress-limit-number> max-suppress
<max-suppress-minutes>} | policy-filter [<policy-name> | none]}
```

To enable route flap dampening for a BGP peer group, use the following command:

```
configure bgp peer-group <peer-group-name> {address-family [ipv4-unicast |
ipv4-multicast]} dampening {{half-life <half-life-minutes> {reuse-limit
<reuse-limit-number> supress-limit <suppress-limit-number> max-suppress
<max-suppress-minutes>}} | policy-filter [<policy-name> | none]}
```

You can supply the dampening parameters directly through the CLI command, or use the command to associate a policy that contains the desired parameters.

Disabling Route Flap Dampening

To disable route flap dampening for a BGP neighbor (disabling the dampening will also delete all the configured dampening parameters), use the following command:

```
configure bgp neighbor [<remoteaddr> | all] {address-family [ipv4-unicast |
ipv4-multicast]} no-dampening
```

To disable route flap dampening for a BGP peer group, use the following command:

```
configure bgp peer-group <peer-group-name> no-dampening
```

Viewing the Route Flap Dampening Configuration

To view the configured values of the route flap dampening parameters for a BGP neighbor, use the following command:

```
show bgp neighbor <remoteaddr> {address-family [ipv4-unicast | ipv4-multicast]}
[accepted-routes | flap-statistics | received-routes | rejected-routes |
suppressed-routes | transmitted-routes] {detail} [all | as-path <path-expression> |
community [no-advertise | no-export | no-export-subconfed | number <community_num> |
<AS_Num>:<Num> ] | network <ip_addr>/<mask_len> ]
```

To view the configured values of the route flap dampening parameters for a BGP peer group, use the following command:

```
show bgp peer-group {detail | <peer-group-name> {detail}}
```

BGP Route Selection

BGP will select routes based on the following precedence (from highest to lowest):

- higher weight
- higher local preference

- shortest length (shortest AS path)
- lowest origin code
- lowest MED
- route from external peer
- lowest cost to Next Hop
- lowest routerID

Stripping Out Private AS Numbers from Route Updates

Private AS numbers are AS numbers in the range 64512 through 65534. You can remove private AS numbers from the AS path attribute in updates that are sent to external BGP (EBGP) neighbors. Possible reasons for using private AS numbers include:

- The remote AS does not have officially allocated AS numbers.
- You want to conserve AS numbers if you are multi-homed to the local AS.

Private AS numbers should not be advertised on the Internet. Private AS numbers can only be used locally within an administrative domain. Therefore, when routes are advertised out to the Internet, the routes can be stripped out from the AS paths of the advertised routes using this feature.

To configure private AS numbers to be removed from updates, use the following command:

```
enable bgp neighbor [<remoteaddr> | all] remove-private-AS-numbers
```

To disable this feature, use the following command:

```
disable bgp neighbor [<remoteaddr> | all] remove-private-AS-numbers
```

Route Re-Distribution

BGP, OSPF, and RIP can be enabled simultaneously on the switch. Route re-distribution allows the switch to exchange routes, including static, direct, and VIP routes, between any two routing protocols.

Exporting routes from OSPF to BGP, and from BGP to OSPF, are discreet configuration functions. To run OSPF and BGP simultaneously, you must first configure both protocols and then verify the independent operation of each. Then you can configure the routes to export from OSPF to BGP and the routes to export from BGP to OSPF.

Configuring Route Re-Distribution

Exporting routes between any two routing protocols are discreet configuration functions. For example, you must configure the switch to export routes from OSPF to BGP and, if desired, you must configure the switch to export routes from BGP to OSPF. You must first configure both protocols and then verify the independent operation of each. Then you can configure the routes to export from OSPF to BGP and the routes to export from BGP to OSPF.

You can use route maps to associate BGP attributes including Community, NextHop, MED, Origin, and Local Preference with the routes. Route maps can also be used to filter out exported routes.

To enable or disable the exporting of OSPF, IS-IS, RIP, static, direct (interface), and VIP routes to BGP, use the following commands:

```
enable bgp export [direct | ospf | ospf-extern1 | ospf-extern2 | ospf-inter |
ospf-intra | rip | static] {address-family [ipv4-unicast | ipv4-multicast]}
{export-policy <policy-name>}

disable bgp export [direct | ospf | ospf-extern1 | ospf-extern2 | ospf-inter |
ospf-intra | rip | static] {address-family [ipv4-unicast | ipv4-multicast]}
```

Using the `export` command to redistribute routes complements the redistribution of routes using the `configure bgp add network` command. The `configure bgp add network` command adds the route to BGP only if the route is present in the routing table. The `enable bgp export` command redistributes an individual route from the routing table to BGP. If you use both commands to redistribute routes, the routes redistributed using the `network` command take precedence over routes redistributed using the `export` command.

BGP Static Network

ExtremeWare XOS BGP allows users to add static networks in BGP, which will be redistributed (advertised) into the BGP domain if there is a corresponding active route in the IP routing table. Users can associate a policy with the static BGP network to change or set the route attributes before it would be advertised to the BGP neighbors.

Use the following command to create a static BGP network:

```
configure bgp add network {address-family [ipv4-unicast | ipv4-multicast]}
<ipaddr>/<mask_len> {network-policy <policy>}
```

Use the following command to delete a static BGP network

```
configure bgp delete network {address-family [ipv4-unicast | ipv4-multicast]} [all |
<ipaddress>]
```

15 IP Multicast Routing

This chapter covers the following topics:

- Overview on page 217
  - PIM Overview on page 218
  - PIM Overview on page 218
  - IGMP Overview on page 219
- Configuring IP Multicasting Routing on page 220
- Configuration Examples on page 221

For more information on IP multicasting, refer to the following publications:

- RFC 1112 – *Host Extension for IP Multicasting*
- RFC 2236 – *Internet Group Management Protocol, Version 2*
- PIM-DM Version 2 – *draft_ietf_pim_v2_dm_03*
- RFC 2362 – *PIM-SM Version 2*

The following URLs point to the Web sites for the IETF Working Groups:

IEFT PIM Working Group:

http://www.ietf.org/html.charters/pim-charter.html

Overview

IP multicast routing is a function that allows a single IP host to send a packet to a group of IP hosts. This group of hosts can include devices that reside on or outside the local network, or within or across a routing domain.

IP multicast routing consists of the following functions:

- A router that can forward IP multicast packets
- A router-to-router multicast routing protocol (for example, Protocol Independent Multicast (PIM))
- A method for the IP host to communicate its multicast group membership to a router (for example, Internet Group Management Protocol (IGMP))

NOTE

You should configure IP unicast routing before you configure IP multicast routing.

PIM Overview

The switch supports both dense mode and sparse mode operation. You can configure dense mode or sparse mode on a per-interface basis. After they are enabled, some interfaces can run dense mode, while others run sparse mode.

PIM Dense Mode

Protocol Independent Multicast-Dense Mode (PIM-DM) is a multicast routing protocol. PIM-DM routers perform reverse path multicasting (RPM). However, instead of exchanging its own unicast route tables for the RPM algorithm, PIM-DM uses the existing unicast routing table for the reverse path. As a result, PIM-DM requires less system memory.

PIM-DM is a broadcast and prune protocol, which allows you to prune and graft multicast routes.

PIM Sparse Mode (PIM-SM)

Unlike PIM-DM, PIM-SM is an explicit join and prune protocol, and it supports shared trees as well as shortest path trees (SPTs). The routers must explicitly join the group(s) in which they are interested in becoming a member, which is beneficial for large networks that have group members who are sparsely distributed.

Using PIM-SM, the router sends a join message to the rendezvous point (RP). The RP is a central multicast router that is responsible for receiving and distributing multicast packets. By default, the user configured RP is dynamic. You can also define a static RP in your network.

When a router has a multicast packet to distribute, it encapsulates the packet in a unicast message and sends it to the RP. The RP decapsulates the multicast packet and distributes it among all member routers.

When a router determines that the multicast rate has exceeded a configured threshold, that router can send an explicit join to the originating router. When this occurs, the receiving router gets the multicast directly from the sending router, and bypasses the RP.

NOTE

You can run either PIM-DM or PIM-SM per VLAN.

PIM Mode Interoperation

An Extreme Networks switch can function as a PIM multicast border router (PMBR). A PMBR integrates PIM-SM and PIM-DM traffic.

When forwarding PIM-DM traffic into a PIM-SM network, the PMBR notifies the RP that the PIM-DM network exists. The PMBR forwards PIM-DM multicast packets to the RP, which, in turn, forwards the packets to those routers that have joined the multicast group.

The PMBR also forwards PIM-SM traffic to a PIM-DM network, based on the (*.*.RP) entry. The PMBR sends a join message to the RP and the PMBR forwards traffic from the RP into the PIM-DM network.

No commands are required to enable PIM mode interoperation. PIM mode interoperation is automatically enabled when a dense mode interface and a sparse mode interface are enabled on the same switch.

IGMP Overview

IGMP is a protocol used by an IP host to register its IP multicast group membership with a router. Periodically, the router queries the multicast group to see if the group is still in use. If the group is still active, a single IP host responds to the query, and group registration is maintained.

IGMP is enabled by default on the switch. However, the switch can be configured to disable the generation of periodic IGMP query packets. IGMP should be enabled when the switch is configured to perform IP unicast or IP multicast routing.

IGMP Snooping

IGMP snooping is a layer 2 function of the switch. It does not require multicast routing to be enabled. In IGMP snooping, the layer 2 switch keeps track of IGMP requests, and only forwards multicast traffic to the part of the local network that requires it. IGMP snooping optimizes the usage of network bandwidth, and prevents multicast traffic from being flooded to parts of the local network that do not need it.

IGMP snooping is enabled by default on the switch. If IGMP snooping is disabled, all IGMP and IP multicast traffic floods within a given VLAN. IGMP snooping expects at least one device on every VLAN to periodically generate IGMP query messages. The static IGMP snooping entries do not require periodic query, but do require a query in order to retrieve them after the `clear igmp snooping` command.

When a port sends an IGMP leave message, the switch removes the IGMP snooping entry after 1000 milliseconds (the leave time is configurable, ranging from 0 to 10000 ms). The switch sends a query to determine which ports want to remain in the multicast group. If other members of the VLAN want to remain in the multicast group, the router ignores the leave message, but the port that requests removal is removed from the IGMP snooping table.

If the last port within a VLAN sends an IGMP leave message and the router does not receive any responses to the query, then the router immediately removes the VLAN from the multicast group.

Static IGMP

To receive multicast traffic, a host needs to explicitly join a multicast group by sending an IGMP report, then the traffic is forwarded to that host. There are situations where you would like multicast traffic to be forwarded to a port where a multicast enabled host is not available (for example, testing multicast configurations). Static IGMP emulates a host or router attached to a switch port, so that multicast traffic will be forwarded to that port. To emulate a host so as to forward a particular multicast group to a port, emulate a router to forward all multicast groups to a port. Use the following command to emulate a host on a port:

```
configure igmp snooping vlan <vlan name> ports <portlist> add static group <group address>
```

To emulate a multicast router on a port, use the following command:

```
configure igmp snooping {vlan} <vlanname> ports <portlist> add static router
```

To remove these entries, use the corresponding command:

```
configure igmp snooping {vlan} <vlanname> ports <portlist> delete static
group [<ip_address> | all]
configure igmp snooping vlan <vlanname> ports <portlist> delete static router
```

To display the IGMP snooping static groups, use the following command:

```
show igmp snooping vlan <name> static [group | router]
```

IGMP Snooping Filters

IGMP snooping filters allow you to configure a policy file on a port to allow or deny IGMP report and leave packets coming into the port. For details on creating policy files, see “Management Access Security” on page 129. After you have created an policy file, use the following command to associate the policy file and filter a set of ports:

```
configure igmp snooping vlan <vlan name> ports <portlist> filter <policy file>
```

To remove the filter, use the following command:

```
configure igmp snooping vlan <vlan name> ports <portlist> filter none
```

To display the IGMP snooping filters, use the following command:

```
show igmp snooping {vlan} <name> filter
```

Configuring IP Multicasting Routing

To configure IP multicast routing:

1. Configure the system for IP unicast routing.
2. Enable multicast routing on the interface using the following command:

   ```
   enable ipmcforwarding {vlan <name>}
   ```
3. Enable PIM on all IP multicast routing interfaces using the following command:

   ```
   configure pim add vlan [<vlan_name> | all] {dense | sparse}
   ```
4. Enable PIM on the router using the following command:

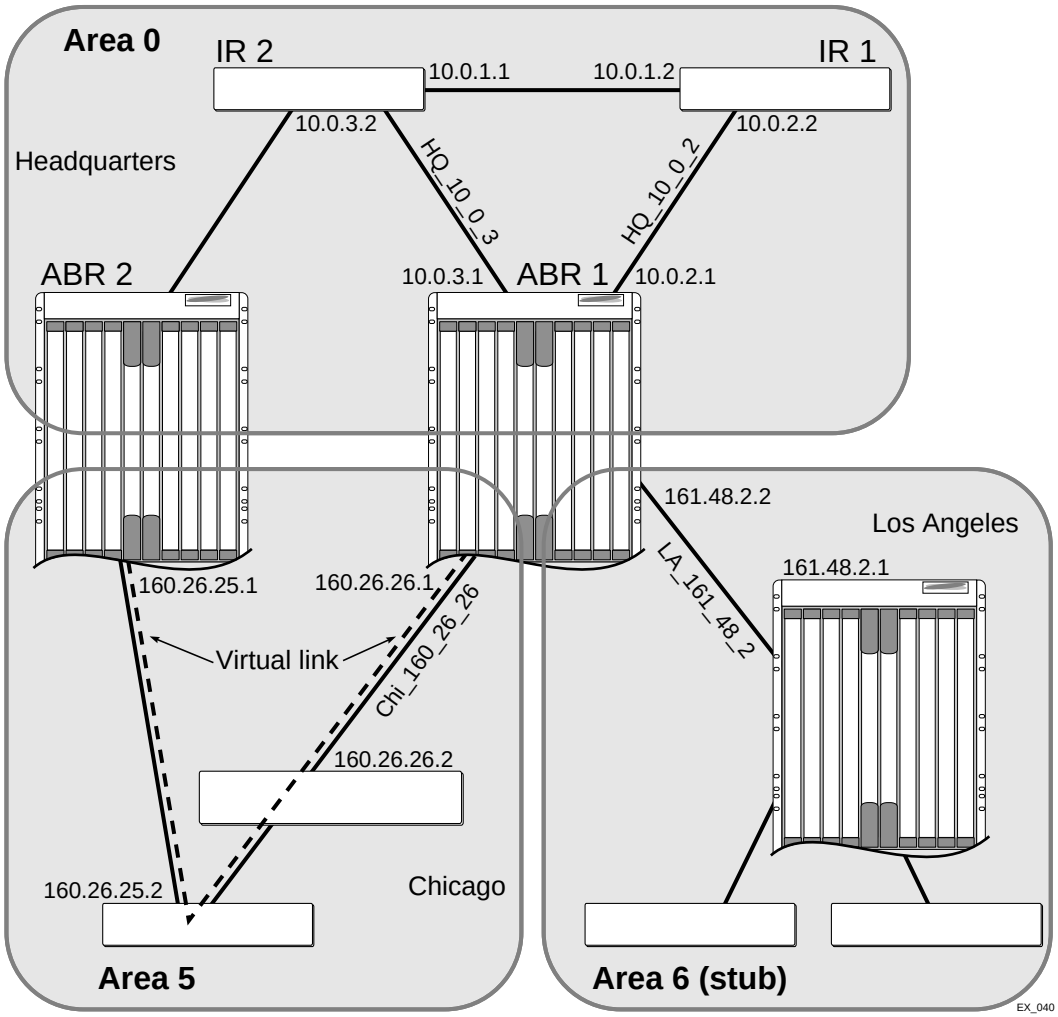
   ```
   enable pim
   ```

Configuration Examples

Figure 40 and Figure 41 are used in Chapter 13 to describe the OSPF configuration on a switch. Refer to Chapter 13 for more information about configuring OSPF. In Figure 40, the system labeled IR1 is configured for IP multicast routing, using PIM-DM. In Figure 41, the system labeled ABR1 is configured for IP multicast routing using PIM-SM.

PIM-DM Configuration Example

Figure 40: IP multicast routing using PIM-DM configuration example



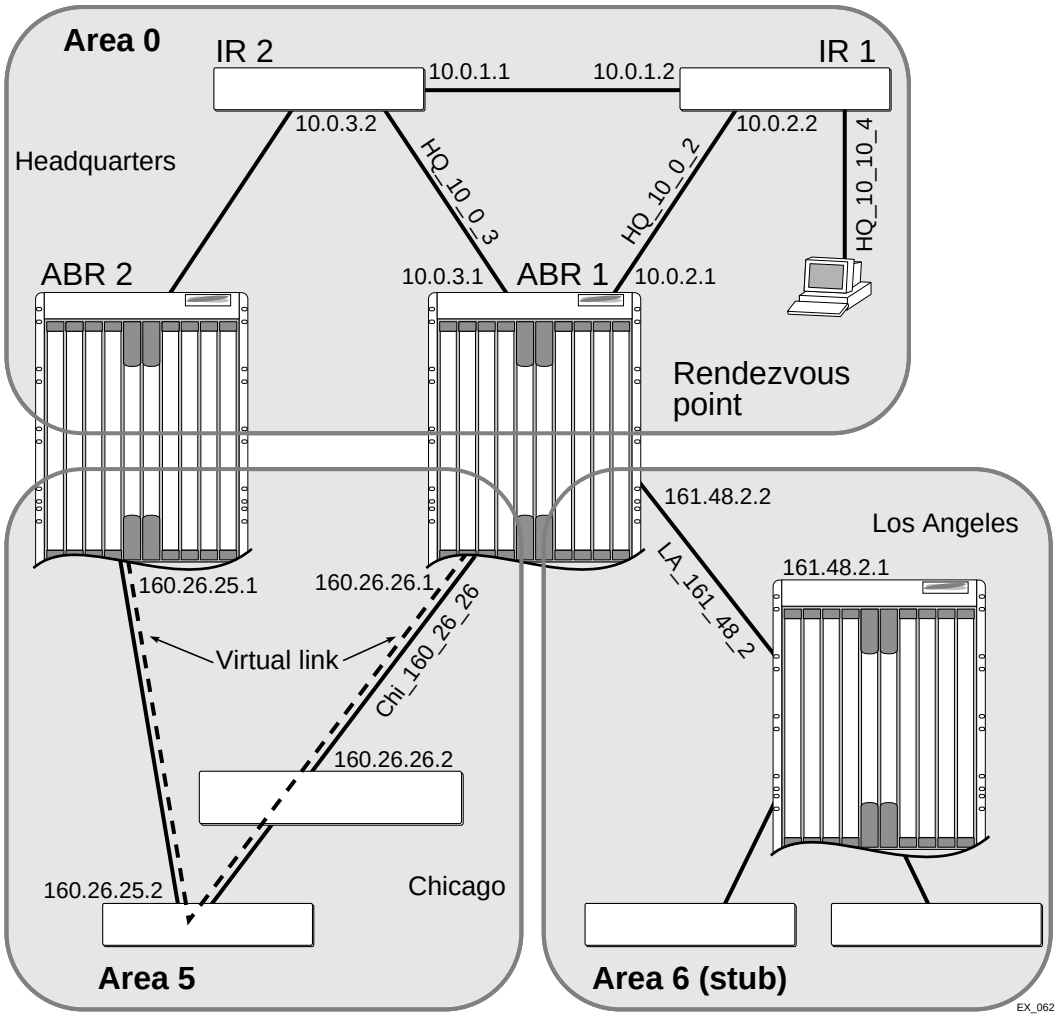
Configuration for IR1

The router labeled IR1 has the following configuration:

```
configure vlan HQ_10_0_1 ipaddress 10.0.1.2 255.255.255.0
configure vlan HQ_10_0_2 ipaddress 10.0.2.2 255.255.255.0
configure ospf add vlan all area 0.0.0.0
enable ipforwarding
enable ospf
enable ipmcforwarding
configure pim add vlan all dense
enable pim
```

The following example configures PIM-SM.

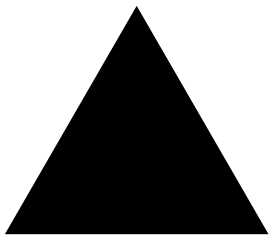
Figure 41: IP multicast routing using PIM-SM configuration example



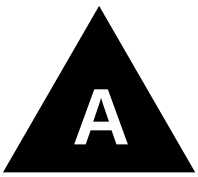
Configuration for ABR1

The router labeled ABR1 has the following configuration:

```
configure vlan HQ_10_0_2 ipaddress 10.0.2.1 255.255.255.0
configure vlan HQ_10_0_3 ipaddress 10.0.3.1 255.255.255.0
configure vlan LA_161_48_2 ipaddress 161.48.2.2 255.255.255.0
configure vlan CHI_160_26_26 ipaddress 160.26.26.1 255.255.255.0
configure ospf add vlan all area 0.0.0.0
enable ipforwarding
enable ipmcforwarding
configure pim add vlan all sparse
enable loopback HQ_10_0_3
tftp TFTP_SERV -g -r rp_list.pol
configure pim crp HQ_10_0_3 rp-list 30
configure pim cbsr HQ_10_0_3 30
```



Part 3
Appendixes



Software Upgrade and Boot Options

This appendix describes the following topics:

- Downloading a New Image on page 227
- Saving Configuration Changes on page 229
- Using TFTP to Upload the Configuration on page 231
- Using TFTP to Download the Configuration on page 232
- Accessing the Bootloader on page 232

Downloading a New Image

The image file contains the executable code that runs on the switch. It comes preinstalled from the factory. As new versions of the image are released, you should upgrade the software running on your system.

The image is upgraded by using a download procedure from either a Trivial File Transfer Protocol (TFTP) server on the network or from a PC connected to the serial port using the XMODEM protocol. Downloading a new image involves the following steps:

- Loading the new image onto a TFTP server on your network (if you will be using TFTP).
- Loading the new image onto a PC (if you will be using XMODEM).
- Selecting the partition to use when downloading an image. For more information, see “Selecting a Primary or a Secondary Image” on page 228.
- Downloading the new image to the switch using the following command:

  ```
  download image [<hostname> | <ipaddress>] <filename> {[{vr} <vrid>]}
  ```

  Where the following is true:

  `hostname`—Is the hostname of the TFTP server. (You must enable DNS to use this option.)

  `ipaddress`—Is the IP address of the TFTP server.

  `filename`—Is the filename of the new image.

  `vrid`—Is the name of the virtual router.

Before the download begins, you are asked if you want to install the image immediately after the download is finished. If you install the image immediately after download, you must reboot the switch. Enter y to install the image after download. Enter n to install the image at a later time.

If you download and install the software image on the active partition, you need to reboot the switch. The following message appears when downloading and installing on the active partition:

```
Image will be installed to the active partition, a reboot required. Do you want to
continue? (y or n)
```

Enter y to continue the installation and reboot the switch. Enter n to cancel.

If you choose to install the image at a later time, use the following command to install the software:

```
install image <fname> {reboot}
```

where fname specifies the filename of the new, downloaded image.

Selecting a Primary or a Secondary Image

The switch comes with one image preinstalled from the factory and can store can store up to two images: a primary and a secondary. When downloading a new image, you select which partition (primary or secondary) to install the new image. If you do not specify a partition, the software image is downloaded and installed into the current (active) partition. If you want to install the software image to the alternate partition, you must specify that partition before downloading the image. To view your current (active) partition, use the following command:

```
show switch
```

Output from this command includes the selected and booted images and if they are in the primary or secondary partition.

If two MSMs are installed in the BlackDiamond 10808 switch, the downloaded image is saved to the same location on each one.

You can select which image the switch will load on the next reboot by using the following command:

```
use image {partition} <partition>
```

Understanding the Image Version String

The image version string contains build information for each version of ExtremeWare XOS. You can use either the show version or show switch command to display the ExtremeWare XOS version running on your switch.

Depending on the CLI command, the output is structured as follows:

- show version

  ExtremeWare XOS Version <major>.<minor>.<patch>.<build>

  For example: ExtremeWare XOS version 10.1.0.86
- show switch

  <major>.<minor>.<patch>.<build>

  For example: 10.1.0.86

Table 38 describes the image version fields.

Table 38: Image version fields

Field	Description
major	Specifies the ExtremeWare XOS Major version number.
minor	Specifies the ExtremeWare XOS Minor version number.
patch	Identifies a specific patch release.
build	Specifies the ExtremeWare XOS build number. This value is reset to zero for each new Major and Minor release.

Software Signatures

Each ExtremeWare XOS image contains a unique signature. The BootROM checks for signature compatibility and denies an incompatible software upgrade. In addition, the software checks both the installed BootROM and software and also denies an incompatible upgrade.

Rebooting the Switch

To reboot the switch immediately, use the following command:

```
reboot {time <date> <time> | cancel} {slot <slot number> | msm <slotid>}
```

where `date` is the date and `time` is the time (using a 24-hour clock format) when the switch will be rebooted. The values use the following format:

```
mm/dd/yyyy hh:mm:ss
```

If you do not specify a reboot time, the reboot occurs immediately following the command, and any previously schedule reboots are cancelled. To cancel a previously scheduled reboot, use the `cancel` option.

Saving Configuration Changes

The configuration is the customized set of parameters that you have selected to run on the switch. As you make configuration changes, the new settings are stored in run-time memory. Settings that are stored in run-time memory are not retained by the switch when the switch is rebooted. To retain the settings, and have them loaded when you reboot the switch, you must save the configuration to nonvolatile storage.

The switch can store multiple user-defined configuration files, each with its own file name. By default, the switch has two pre-named configurations: a primary and a secondary configuration. When you save configuration changes, you can select to which configuration you want the changes saved or you can save the changes to a new configuration file. If you do not specify a file name, the changes are saved to the configuration file currently in use. Or if you have never saved any configurations, you are asked to save your changes to the primary configuration.

NOTE

Configuration files have a .cfg file extension. When you enter the name of the file in the CLI, the system automatically adds the .cfg file extension.

If you have made a mistake, or you must revert to the configuration as it was before you started making changes, you can tell the switch to use the backup configuration on the next reboot.

To save the configuration, use the following command:

```
save configuration {primary | secondary | <existing-config> | <new-config>}
```

Where the following is true:

- `primary`—Specifies the primary saved configuration
- `secondary`—Specifies the secondary saved configuration
- `existing-config`—Specifies an existing user-defined configuration (displays a list of available user-defined configuration files)
- `new-config`—Specifies a new user-defined configuration

You are then prompted to save the changes. Enter `y` to save the changes or `n` to cancel the process.

To use the configuration, use the following command:

```
use configuration [primary | secondary | <file_name>
```

Where the following is true:

- `primary`—Specifies the primary saved configuration
- `secondary`—Specifies the secondary saved configuration
- `file_name`—Specifies an existing user-defined configuration (displays a list of available user-defined configuration files)

The configuration takes effect on the next reboot.

NOTE

If the switch is rebooted while in the middle of a configuration save, the switch boots to factory default settings if the previously saved configuration file is overwritten. The configuration that is not in the process of being saved is unaffected.

Returning to Factory Defaults

To return the switch configuration to factory defaults, use the following command:

```
unconfigure switch
```

This command resets the entire configuration, with the exception of user accounts and passwords that have been configured, and the date and time.

To erase the currently selected configuration image, reset all switch parameters, and reboot the switch, use the following command:

```
unconfigure switch {all}
```

Using TFTP to Upload the Configuration

You can upload the current configuration to a TFTP server on your network. The uploaded configuration file retains your system configuration and is saved in XML format. This allows you to send a copy of the configuration file to the Extreme Networks Technical Support department for problem-solving purposes.

You are unable to view XOS configuration files with a text editor. To view your current switch configuration, use the `show` commands available on your switch. For more information about the `show` commands, see the *ExtremeWare XOS Command Reference Guide*.

To upload the configuration, use the following command:

```
tftp <ip_address> -p -l <local_file>
```

Where the following is true:

- `ip_address`—Is the IP address of the TFTP server
- `-p`—Puts the specified file from the local host and copies it to the TFTP server
- `local_file`—Specifies the name of the configuration file that you want to save to the TFTP server

If you upload a configuration file and see the following message:

```
Error: No such file or directory
```

Check to make sure that you entered the file name correctly, including the .cfg extension, and that you entered the correct IP address for the TFTP server.

Displaying Configuration Files

To see a list of the configuration files in your switch, use the following command:

```
ls
```

You can also see a complete list of configuration files by entering the following syntax followed by the Tab key:

- `save configuration`
- `use configuration`

Renaming Configuration Files

To rename an existing configuration file in your system, use the following command:

```
mv <old-name> <new-name>
```

Where the following is true:

- `old-name`—Specifies the current name of the configuration file
- `new-name`—Specifies the new name of the configuration file

If you rename a configuration file, make sure the new file name has the same .cfg extension.

Deleting Configuration Files

To delete a configuration file from your system, use the following command:

```
rm <file-name>
```

Where `file-name` specifies the name of the configuration file to delete.

After you delete a file from the system, it is no longer available.

Using TFTP to Download the Configuration

You can download previously saved XML formatted XOS configuration files from a TFTP host to the switch to modify the switch configuration. To download the configuration, use the following command:

```
tftp <ip_address> -g -r <remote_file>
```

Where the following is true:

- `ip_address`—Is the IP address of the TFTP server
- `-g`—Gets the specified file from the TFTP server and copies it to the local host
- `remote_file`—Specifies the name of the configuration file that you want to retrieve from the TFTP server

If you download a configuration file and see the following message:

```
Error: Transfer timed out
```

Check to make sure that you entered the file name correctly, including the .cfg extension, and that you entered the correct IP address for the TFTP server.

Configurations are downloaded and saved into switch non-volatile memory. The configuration is applied after you reboot the switch.

If the configuration currently running in the switch does not match the configuration that the switch used when it originally booted, an asterisk (*) appears before the command line prompt when using the CLI.

Accessing the Bootloader

The Bootloader of the switch initializes certain important switch variables during the boot process. In the event the switch does not boot properly, some boot option functions can be accessed through the Bootloader.

Interaction with the Bootloader is only required under special circumstances, and should be done only under the direction of Extreme Networks Customer Support. The necessity of using these functions implies a non-standard problem which requires the assistance of Extreme Networks Customer Support.

To access the Bootloader, follow these steps:

1. Attach a serial cable to the console port of the switch.
2. Attach the other end of the serial cable to a properly configured terminal or terminal emulator, power cycle the switch and depress any ASCII key on the keyboard of the terminal during the boot up process.

   NOTE

   To access the Bootloader, you can depress any key until the applications load and run on the switch.

   As soon as you see the `BOOTLOADER->` prompt, release the key. You can issue a series of commands to:

   — View the installed images

   — Select the image to boot from

   — Select the configuration to use

   To see a list of available commands or additional information about a specific command, enter `h` or type `help`.

The following describes some ways that you can use the bootloader.

- Viewing images—To display a list of installed images, use the `show images` command.
- Selecting an image—To change the image that the switch boots from in flash memory, use the `boot {image name}` command. If you specify `image name`, the specified image is booted. If you do not specify an image name, the default image is booted.
- Selecting a configuration—To select a different configuration from the one currently running, use the `config {default | file <filename> | none}` command. This command is useful if you experience a problem with the current configuration and there is an alternate configuration available.

  — `file`—Specifies a configuration file name

  — `default`—Specifies the default configuration file

  — `none`—Uses no configuration

  To view the current configuration, use this command without any arguments.

To exit the Bootloader, use either the `boot` or `boot [1-4]` command. Specifying `boot` without any arguments runs the currently selected XOS image. Specifying boot with arguments will either run a newly selected XOS image or run diagnostics on the MSM. For more information about the `boot [1-4]` command and running diagnostics on the MSM, see “Running Diagnostics on MSM Modules” on page 92.

B Troubleshooting

If you encounter problems when using the switch, this appendix may be helpful. If you have a problem not listed here or in the release notes, contact your local technical support representative.

LEDs

Power LED does not light:

Check that the power cable is firmly connected to the device and to the supply outlet.

On powering-up, the MGMT LED lights yellow:

The device has failed its Power On Self Test (POST) and you should contact your supplier for advice.

A link is connected, but the Status LED does not light:

Check that:

- All connections are secure.
- Cables are free from damage.
- The devices at both ends of the link are powered-up.
- Both ends of the Gigabit link are set to the same autonegotiation state.

  The Gigabit link must be enabled or disabled on both sides. If the two sides are different, typically the side with autonegotiation disabled will have the link LED lit, and the side with autonegotiation enabled will not be lit. The default configuration for a Gigabit port is autonegotiation enabled. This can be verified by entering the following command:

  ```
  show port configuration
  ```

On power-on, some I/O modules do not boot:

Check if you are using 110V power input. The BlackDiamond switch powers only up to four modules if it is connected to a 110V outlet.

Error LED on the MSM turns amber:

Check the syslog message for a “critical” software errors.

Status LED on the I/O module turns amber:

Check the syslog message for a related I/O module error. If the error is an inserted an I/O module that conflicts with the software configuration, use one of the following commands to reset the slot configuration:

```
clear slot
```

```
configure slot <slot> module <module_type>
```

Otherwise, contact Extreme Networks for further assistance.

ENV LED on the MSM turns amber:

Check each of the power supplies and all of the fans. Additionally, the status of these is indicated in the `show powersupplies` and `show powersupplies` displays.

Switch does not power up:

All products manufactured by Extreme Networks use digital power supplies with surge protection. In the event of a power surge, the protection circuits shut down the power supply. To reset, unplug the switch for 1 minute, plug it back in, and attempt to power up the switch.

If this does not work, try using a different power source (different power strip/outlet) and power cord.

Using the Command Line Interface

The initial welcome prompt does not display:

Check that your terminal or terminal emulator is correctly configured.

For console port access, you may need to press [Return] several times before the welcome prompt appears.

Check the settings on your terminal or terminal emulator. The settings are 9600 baud, 8 data bits, 1 stop bit, no parity, XON/OFF flow control enabled.

The SNMP Network Manager cannot access the device:

Check that the device IP address, subnet mask, and default router are correctly configured, and that the device has been reset.

Check that the device IP address is correctly recorded by the SNMP Network Manager (refer to the user documentation for the Network Manager).

Check that the community strings configured for the system and Network Manager are the same.

Check that the SNMPv3 USM, Auth, and VACM configured for the system and Network Manager are the same.

Check that SNMP access was not disabled for the system.

The Telnet workstation cannot access the device:

Check that the device IP address, subnet mask and default router are correctly configured, and that the device has been reset. Ensure that you enter the IP address of the switch correctly when invoking the Telnet facility. Check that Telnet access was not disabled for the switch. If you attempt to log in and the maximum number of Telnet sessions are being used, you should receive an error message indicating so.

Traps are not received by the SNMP Network Manager:

Check that the SNMP Network Manager's IP address and community string are correctly configured, and that the IP address of the Trap Receiver is configured properly on the system.

The SNMP Network Manager or Telnet workstation can no longer access the device:

Check that Telnet access or SNMP access is enabled.

Check that the port through which you are trying to access the device has not been disabled. If it is enabled, check the connections and network cabling at the port.

Check that the port through which you are trying to access the device is in a correctly configured VLAN.

Try accessing the device through a different port. If you can now access the device, a problem with the original port is indicated. Re-examine the connections and cabling.

A network problem may be preventing you accessing the device over the network. Try accessing the device through the console port.

Check that the community strings configured for the device and the Network Manager are the same.

Check that SNMP access was not disabled for the system.

Permanent entries remain in the FDB:

If you have made a permanent entry in the FDB (which requires you to specify the VLAN to which it belongs and then delete the VLAN), the FDB entry will remain. Though causing no harm, you must manually delete the entry from the FDB if you want to remove it.

Default and Static Routes:

If you have defined static or default routes, those routes will remain in the configuration independent of whether the VLAN and VLAN IP address that used them remains. You should manually delete the routes if no VLAN IP address is capable of using them.

You forget your password and cannot log in:

If you are not an administrator, another user having administrator access level can log in, delete your user name, and create a new user name for you, with a new password.

Alternatively, another user having administrator access level can log in and initialize the device. This will return all configuration information (including passwords) to the initial values.

In the case where no one knows a password for an administrator level user, contact your supplier.

Port Configuration

No link light on 10/100 Base port:

If patching from a hub or switch to another hub or switch, ensure that you are using a CAT5 cross-over cable. This is a CAT5 cable that has pins 1&2 on one end connected to pins 3&6 on the other end.

Excessive RX CRC errors:

When a device that has auto-negotiation disabled is connected to an Extreme switch that has auto-negotiation enabled, the Extreme switch links at the correct speed, but in half duplex mode. The Extreme switch 10/100 physical interface uses a method called *parallel detection* to bring up the link. Because the other network device is not participating in auto-negotiation (and does not advertise its capabilities), parallel detection on the Extreme switch is only able to sense 10Mbps versus 100Mbps speed, and not the duplex mode. Therefore, the switch establishes the link in half duplex mode using the correct speed.

The only way to establish a full duplex link is to either force it at both sides, or run auto-negotiation on both sides (using full duplex as an advertised capability, which is the default setting on the Extreme switch).

NOTE

A mismatch of duplex mode between the Extreme switch and another network device will cause poor network performance. Viewing statistics using the `show ports rxerrors` *command on the Extreme switch may display a constant increment of CRC errors. This is characteristic of a duplex mismatch between devices. This is NOT a problem with the Extreme switch.*

Always verify that the Extreme switch and the network device match in configuration for speed and duplex.

No link light on Gigabit fiber port:

Check to ensure that the transmit fiber goes to the receive fiber side of the other device, and vice-versa. All gigabit fiber cables are of the cross-over type.

The Extreme switch has auto-negotiation set to on by default for gigabit ports. These ports need to be set to auto off (using the command `configure port <port #> auto off`) if you are connecting it to devices that do not support auto-negotiation.

Ensure that you are using multi-mode fiber (MMF) when using a 1000BASE-SX GBIC, and single mode fiber (SMF) when using a 1000BASE-LX GBIC. 1000BASE-SX does not work with SMF. 1000BASE-LX works with MMF, but requires the use of a mode conditioning patchcord (MCP).

VLANs

You cannot add a port to a VLAN:

If you attempt to add a port to a VLAN and get an error message similar to

```
localhost:7 # configure vlan marketing add port 1:1,1:2
Failed: Protocol conflict when adding untagged port 1:1,1:2. Either add this port as
tagged or assign another protocol to this VLAN.
```

you already have a VLAN using untagged traffic on a port. Only one VLAN using untagged traffic can be configured on a single physical port.

VLAN configuration can be verified by using the following command:

```
show vlan {<vlan_name> | stpd}
```

The solution for this error is to remove ports 1 and 2 from the VLAN currently using untagged traffic on those ports. If this were the “default” VLAN, the command would be:

```
localhost:23 # configure vlan default del port 1:1,1:2
```

Which should now allow you to re-enter the previous command without error as follows:

```
localhost:26 # configure vlan red add port 1:1,1:2
```

VLAN names:

There are restrictions on VLAN names. They cannot contain whitespaces and cannot start with a numeric value unless you use quotation marks around the name. If a name contains whitespaces, starts with a number, or contains non-alphabetical characters, you must use quotation marks whenever referring to the VLAN name.

802.1Q links do not work correctly:

Remember that VLAN names are only locally significant through the command-line interface. For two switches to communicate across a 802.1Q link, the VLAN ID for the VLAN on one switch should have a corresponding VLAN ID for the VLAN on the other switch.

If you are connecting to a third-party device and have checked that the VLAN IDs are the same, the Ethertype field used to identify packets as 802.1Q packets may differ between the devices. The default value used by the switch is 8100. If the third-party device differs from this and cannot be changed, you may change the 802.1Q Ethertype used by the switch with the following command:

```
configure dot1q ethertype <value>
```

Changing this parameter changes how the system recognizes all tagged frames received, as well as the value it inserts in all tagged frames it transmits.

VLANs, IP Addresses and default routes:

The system can have an IP address for each configured VLAN. It is necessary to have an IP address associated with a VLAN if you intend to manage (Telnet, SNMP, ping) through that VLAN or route IP traffic. You can also configure multiple default routes for the system. The system first tries the default route with the lowest cost metric.

STP

You have connected an endstation directly to the switch and the endstation fails to boot correctly:

The switch has STP enabled, and the endstation is booting before the STP initialization process is complete. Specify that STP has been disabled for that VLAN, or turn off STP for the switch ports of the endstation and devices to which it is attempting to connect, and then reboot the endstation.

The switch keeps aging out endstation entries in the switch Forwarding Database (FDB):

Reduce the number of topology changes by disabling STP on those systems that do not use redundant paths.

Specify that the endstation entries are static or permanent.

Debug Mode

The Event Management System (EMS) provides a standardized way to filter and store messages generated by the switch. With EMS, you must enable debug mode to display debug information. To enable or disable debug mode for EMS, use the following commands:

```
enable log debug-mode
disable log debug-mode
```

After debug mode has been enabled, you can configure EMS to capture specific debug information from the switch. Details of EMS can be found in Chapter 8, “Status Monitoring and Statistics,” on page 91.

System Health Check

The system health check tests the backplane, the CPU, and I/O modules by periodically forwarding packets and checking for the validity of these packets. If you observe a failure, please contact Extreme Technical Support.

To enable system health check, use the following command:

```
enable sys-health-check slot <slot>
```

System health check is enabled by default.

To disable the system health checker, use the following command:

```
disable sys-health-check slot <slot>
```

System Odometer

Each field replaceable component contains a system odometer counter in EEPROM. You can use the `show odometer` command to see how long an individual component has been in service since it was manufactured. The odometer is supported on the following BlackDiamond 10808 components:

- Master MSM and slave MSM
- Backplane
- All modules

The following is sample output from the `show odometer` command:

```
* BD-PC.1 # show odometer
                          Service  First Recorded
  Field Replaceable Units  seconds  Start Date
------------------------  -------  --------------
```

```
Chassis :                   430200  Nov-13-2003
SLOT  1 :                   431200  Nov-13-2003
SLOT  2 :                   432200  Nov-13-2003
SLOT  3 :                   433200  Nov-13-2003
SLOT  4 :                   434200  Nov-13-2003
SLOT  5 :                   435200  Nov-13-2003
SLOT  6 :                   436200  Nov-13-2003
SLOT  7 :                   437200  Nov-13-2003
SLOT  8 :
SLOT  9 :                   439200  Nov-13-2003
SLOT 10 :                        0  Nov-13-2003
```

Contacting Extreme Technical Support

If you have a network issue that you are unable to resolve, contact Extreme Networks technical support. Extreme Networks maintains several Technical Assistance Centers (TACs) around the world to answer networking questions and resolve network problems. You can contact technical support by phone at:

- (800) 998-2408
- (408) 579-2826

Or by email at:

- support@extremenetworks.com

You can also visit the support website at:

http://www.extremenetworks.com/services/resources/

To download software updates (requires a service contract) and documentation (including a .pdf version of this manual).

C Supported Protocols, MIBs, and Standards

The following is a list of software standards and protocols supported by ExtremeWare XOS.

General Routing and Switching	
RFC 1812 Requirements for IP Version 4 Routers	RFC 793 Transmission Control Protocol
RFC 1519 An Architecture for IP Address Allocation with CIDR	RFC 826 Ethernet Address Resolution Protocol: Or converting network protocol addresses to 48.bit Ethernet address for transmission on Ethernet hardware
RFC 1256 ICMP Router Discovery Messages	RFC 2338 Virtual Router Redundancy Protocol
RFC 783 TFTP Protocol (revision 2)	Draft VRRP spec v2.06 (minor modifications to RFC 2338)
RFC 951 Bootstrap Protocol	IEEE 802.1D-1998 Spanning Tree Protocol
RFC 1542 Clarifications and Extensions for the Bootstrap Protocol	IEEE 802.1W - 2001 Rapid Spanning Tree Protocol
RFC 2131 Dynamic Host Configuration Protocol	Definitions of managed objects for bridges with rapid spanning tree protocol Draft-ietf-bridge-rstpm.b-03.txt
RFC 1122 Requirements for Internet Hosts - Communication Layers	IEEE 802.1Q - 1998 Virtual Bridged Local Area Networks
RFC 768 User Datagram Protocol	
RFC 791 Internet Protocol	
RFC 792 Internet Control Message Protocol	

VLANs	
IEEE 802.1Q VLAN Tagging	Multiple STP domains per VLAN
IEEE 802.3ad Static ConfigPort-based VLANs	Virtual MANs
Protocol-sensitive VLANs	

Quality of Service	
IEEE 802.1D -1998 (802.1p) Packet Priority	RFC 2597 Assured Forwarding PHB Group
RFC 2474 Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers	RFC 2475 An Architecture for Differentiated Service
RFC 2598 An Expedited Forwarding PHB	Layer 1-4, Layer 7 Policy-Based Mapping

RIP	
RFC 1058 Routing Information Protocol	RFC 2453 RIP Version 2

OSPF	
RFC 2328 OSPF Version 2	RFC 1765 OSPF Database Overflow
RFC 1587 The OSPF NSSA Option	RFC 2370 The OSPF Opaque LSA Option

BGP4	
RFC 1771 A Border Gateway Protocol 4 (BGP-4)	RFC 1745 BGP4/IDRP for IP---OSPF Interaction
RFC 1965 Autonomous System Confederations for BGP	RFC 2385 Protection of BGP Sessions via the TCP MD5 Signature Option
RFC 2796 BGP Route Reflection - An Alternative to Full Mesh IBGP	RFC 2439 BGP Route Flap Dampening
RFC 1997 BGP Communities Attribute	MBGP

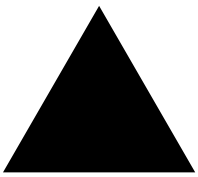
IP Multicast	
RFC 2362 Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification	RFC 2236 Internet Group Management Protocol, Version 2
PIM-DM Draft IETF PIM Dense Mode v2-dm-03)	IGMP Snooping with Configurable Router Registration Forwarding
RFC 1112 Host extensions for IP multicasting	

Management - SNMP & MIBs	
RFC 1157 Simple Network Management Protocol (SNMP)	RFC 3412 Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)
RFC-1215 Convention for defining traps for use with the SNMP	RFC 3413 Simple Network Management Protocol (SNMP) Applications
RFC 1901 Introduction to Community-based SNMPv2	RFC 3414 User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)
RFC 1902 Structure of Management Information for Version 2 of the Simple Network Management Protocol (SNMPv2)	RFC 3415 View-based Access Control Model (VACM) for the Simple Network Management Protocol
RFC 1903 Textual Conventions for Version 2 of the Simple Network Management Protocol (SNMPv2)	ExtremeWare vendor MIB (includes statistics, STP, and others)
RFC 1904 Conformance Statements for Version 2 of the Simple Network Management Protocol (SNMPv2)	RFC-1212 Concise MIB definitions
RFC 1905 Protocol Operations for Version 2 of the Simple Network Management Protocol (SNMPv2)	RFC-1213 Management Information Base for Network Management of TCP/IP-based internets: MIB-II
RFC 1906 Transport Mappings for Version 2 of the Simple Network Management Protocol (SNMPv2)	RFC 2233 Evolution of the Interfaces Group of MIB-II
RFC 1907 Management Information Base for Version 2 of the Simple Network Management Protocol (SNMPv2)	RFC 1724 RIP Version 2 MIB Extension
RFC 1908 Coexistence between Version 1 and Version 2 of the Internet-standard Network Management Framework	RFC 1850 OSPF Version 2 Management Information Base
RFC 3410 Introduction and Applicability Statements for Internet-Standard Management Framework	RFC 1657 Definitions of Managed Objects for the Fourth Version of the Border Gateway Protocol (BGP-4) using SMIv2
RFC 3411 An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks	RFC 2668 Definitions of Managed Objects for IEEE 802.3 Medium Attachment Units (MAUs)
	RFC 2787 Definitions of Managed Objects for the Virtual Router Redundancy Protocol
	IEEE-802.1x MIB

Management - Other	
RFC 854 Telnet Protocol Specification	BSD System Logging Protocol (SYSLOG), with Multiple Syslog Servers
Telnet client and server	Local Messages (criticals stored across reboots)
Configuration logging	RFC 2030 Simple Network Time Protocol (SNTP) Version 4 for IPv4 and OSI
Multiple Images, Multiple Configs	

Security	
Routing protocol authentication (see above)	RFC 2138 Remote Authentication Dial In User Service (RADIUS)
RFC 1492 An Access Control Protocol, Sometimes Called TACACS	RFC 2139 RADIUS Accounting
	Access Control Lists (ACLs)

DiffServ - Standards and MIBs	
RFC 2474 Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers RFC 2475 An Architecture for Differentiated Services	RFC 2597 Assured Forwarding PHB Group RFC 2598 An Expedited Forwarding PHB



Index

Numerics

1d mode, STP 139

A

access control lists 114
access levels 27
access lists, description 113
accounts
  creating 29
  deleting 29
  viewing 29
ACL match conditions 116, 121, 124
ACLs 114
Address Resolution Protocol. *See* ARP
`admin` account 28
aging entries, FDB 76
area 0, OSPF 190
areas, OSPF 189
ARP
  communicating with devices outside subnet 178
  configuring proxy ARP 178
  incapable device 178
  proxy ARP between subnets 178
  proxy ARP, description of 178
  responding to ARP requests 178
  table, displaying 180
atestReceivedEngineTime 42
AuthnoPriv 44
AuthPriv 44
autobind ports 141
autonegotiation, description 55
autonomous system, description 204

B

backbone area, OSPF 190
BGP
  attributes 204
  autonomous system 204
  autonomous system path 204
  cluster 205
  community 205
  description 204
  features 205
  loopback interface 211
  peer groups
    creating 211
    description 211
    mandatory parameters 211
    neighbors 212
  redistributing to OSPF 214
  route aggregation 211
  route reflectors 205
  route selection 213
BlackDiamond switch, port configuration 54
blackhole entries, FDB 77
Bootloader
  accessing 232
  prompt 92
BOOTP relay
  configuring 182
BOOTP, using 36
BootROM, prompt 233
Border Gateway Protocol. *See* BGP

C

carrier vlan 137
CLI
  command shortcuts 22
  line-editing keys 24
  named components 23
  numerical ranges, BlackDiamond switch 23
  symbols 24
  syntax helper 22
  using
command
  syntax, understanding 21
Command-Line Interface. *See* CLI
common commands (table) 25
communicating with devices outside subnet 178
configuration
  downloading 232
  primary and secondary 230
  saving changes 229
  uploading to file 231
console connection 34
conventions
  notice icons, About This Guide 14
  text, About This Guide 14

D

database applications, and QoS 83
database overflow, OSPF 189
default
  gateway 165
  passwords 28
  STP domain 138
  users 28
default VLAN 70
deleting a session 38
DHCP relay, configuring 182
DiffServ, configuring 87
disabling a switch port 54
disabling route advertising (RIP) 187
disconnecting a Telnet session 38
distance-vector protocol, description 186
DNS, description 29
Domain Name Service. *See* DNS
domains, STP 137
dynamic entries, FDB 76
dynamic routes 177

E

EDP, description 60
EMISTP
  description 139
  example 145
  rules 146
enabling a switch port 54
errors, port 94
establishing a Telnet session 35
Extreme Discovery Protocol *See* EDP

F

FDB
  adding an entry 75
  aging entries 76
  blackhole entries 77
  contents 75
  creating a permanent entry example 77
  displaying 78
  dynamic entries 76
  entries 75
  limiting entries 78
  non-aging entries 76
  permanent entries 77
  prioritizing entries 78
file server applications, and QoS 83
flow control 55
Forwarding Database. *See* FDB

G

Greenwich Mean Time Offsets (table) 51
groups 43

I

IEEE 802.1Q 64
IGMP
  description 219
  snooping 219
  static 219
image
  downloading 227
  primary and secondary 228
  upgrading 227
interfaces, router 176
Internet Group Management Protocol. *See* IGMP
IP address, entering 36
IP multicast routing
  configuring 220
  description 19, 217
  example 221
  IGMP
    description 219
    snooping 219
  PIM mode interoperation 218
  PIM multicast border router (PMBR) 218
  PIM-DM 218
  PIM-SM 218
IP unicast routing
  BOOTP relay 182
  configuration examples 180
  configuring 179
  default gateway 175
  description 18
  DHCP relay 182
  enabling 180
  proxy ARP 178
  router interfaces 176
  routing table
    dynamic routes 177
    multiple routes 178
    populating 177
    static routes 177
  verifying the configuration 180
IS-IS
  redistributing routes 194

J

jumbo frames
  description 56
  enabling 56
  IP fragmentation 57
  path MTU discovery 56

K

keys
  line-editing 24
  port monitoring 95

L

line-editing keys 24
link types, configuring in RSTP 150
link-state database 188
link-state protocol, description 186
load sharing
  configuring 58
  description 58
  dynamic 58
  load-sharing group, description 58
  master port 58
  static 58
  verifying the configuration 59
logging in 28

M

MAC-based security 78
management access 27
management port 34
Management Switch Fabric Module. *See* MSM
manually bind ports 140
master port, load sharing 58
match conditions, ACL 116, 121, 124
maximum Telnet session 35
maximum XOS shell sessions 34
mgmt VLAN 35
MIBs 40
modular switch
  configuring load sharing 58
  enabling and disabling ports 54
  jumbo frames 56
  load sharing example 59
  port number 54
  slot configuration 53
  verifying load sharing 59
monitoring the switch 91
MSM 33
multiple routes 178

N

names, VLANs 70
native VLAN, PVST+ 148
noAuthnoPriv 44
node statistics, viewing 100
non-aging entries, FDB 76
Not-So-Stubby_Area. *See* NSSA
NSSA. *See* OSPF

O

opaque LSAs, OSPF 189
Open Shortest Path First. *See* OSPF
opening a Telnet session 35
OSPF
  advantages 186
  area 0 190
  areas 189
  backbone area 190
  configuration example 199
  consistency 189
  database overflow 189
  description 186, 188
  display filtering 201
  link type 193
  link-state database 188
  normal area 191
  NSSA 190
  opaque LSAs 189
  point-to-point links 193
  redistributing routes 194
  redistributing to BGP 214
  router types 190
  settings, displaying 200
  stub area 190
  virtual link 191
  wait interval, configuring 197

P

partition 228
passwords
  default 28
  forgetting 29
path MTU discovery 56
permanent entries, FDB 77
Per-VLAN Spanning Tree. *See* PVST+
PIM
  mode interoperation 218
  multicast border router (PMBR) 218
PIM-DM
  description 218
PIM-SM
  description 218
  rendezvous point 218
`ping` command 30
poison reverse 187
port
  autonegotiation 55
  BlackDiamond switch 54
  configuring on BlackDiamond switch 54
  enabling and disabling 54
  errors,viewing 94
  monitoring display keys 95
  priority, STP 160
  receive errors 94
  statistics, viewing 93
  STP state, displaying 164
  STPD membership 137
  transmit errors 94
port mode 139, 160
port-based VLANs 62
port-mirroring
  description 59
  modular switch example 60
primary image 228
private community, SNMP 40
profiles, QoS 84
protected vlan 137
protocol filters 68
Protocol Independent Multicast- Dense Mode. *See* PIM-DM
Protocol Independent Multicast- Sparse Mode. *See* PIM-SM
protocol-based VLANs 67
proxy ARP
  communicating with devices outside subnet 178
  conditions 178
  configuring 178
  MAC address in response 178
  responding to requests 178
  subnets 178
proxy ARP, description 178
public community, SNMP 40
PVST+
  description 148
  native VLAN 148
  VLAN mapping 148
PVST+ mode 139

Q

QoS
  802.1p priority 86
  applications 82
  database applications 83
  default QoS profiles 85
  description 18, 81

DiffServ, configuring 87
examples
source port 89
file server applications 83
maximum bandwidth 84
minimum bandwidth 84
priority 85
profiles
default 85
description 84
parameters 84
traffic groupings 85
description 84
explicit packet marking 86
source port 89
traffic groupings (table) 85
verifying 90
video applications 83
voice applications 82
web browsing applications 83
QoS monitor
description 90
Quality of Servce. *See* QoS 82

R

RADIUS
and TACACS+ 48, 130, 132
client configuration 131
description 48, 130
RFC 2138 attributes 131
servers 130
TCP port 131
rapid root failover 142
Rapid Spanning Tree Protocol. *See* RSTP
receive errors 94
renaming a VLAN 70
reset to factory defaults 230
responding to ARP requests 178
RIP
advantages 186
configuration example 196
description 186, 187
disabling route advertising 187
enabling 180
limitations 186
poison reverse 187
redistributing routes 194
redistributing to BGP 214
routing table entries 187
split horizon 187
triggered updates 187
version 2 188
route maps, description 119
router interfaces 176
router types, OSPF 190
Routing Information Protocol. *See* RIP
routing table, populating 177
routing. *See* IP unicast routing
RSTP
configuring link types 150
designated port rapid behavior 154
link types 150
auto 150
broadcast 150
edge 150
point-to-point 150
operation 152
overview 148
port roles 149
alternate 149
backup 150
designated 149
edge 150
root 149
propogating topology information 154
receiving bridge behavior 154
root port rapid behavior 153
terms 149
timers 151

S

saving configuration changes 229
secondary image 228
security name 43
sessions, deleting 38
Simple Network Management Protocol. *See* SNMP
slot
automatic configuration 53
clearing 53
manually configuring 53
mismatch 53
SNAP protocol 69
SNMP
community strings 40
configuring 40
settings, displaying 40
supported MIBs 40
system contact 40
system location 40
system name 40
trap receivers 40
using 39
SNMPEngineBoots 42
snmpEngineID 42
SNMPEngineTime 42
SNTP
configuring 49
Daylight Savings Time 49
description 48
example 52
Greenwich Mean Time offset 49
Greenwich Mean Time Offsets (table) 51
NTP servers 49
Spanning Tree Protocol. *See* STP
speed, ports 55
split horizon 187
stand-alone switch
enabling and disabling ports 54
jumbo frames 56
load sharing 58
verifying load sharing 59
static IGMP 219
static routes 177
statistics
node 100
port 93
status monitoring 91

STP
1D mode 139
advanced example 145
and VLANs 137
and VRRP 169
autobind ports 141
basic configuration example 142
bridge priority 160
carrier vlan 137
configurable parameters 160
configuration examples 160
configuring 159
description 18
displaying settings 163
domains 137
EMISTP
description 139
example 145
rules 146
forward delay 160
hello time 160
manually bind ports 140
max age 160
overview 135
path cost 160
port mode 160
port modes 139
port priority 160
port state, displaying 164
port states 140
blocking 140
disabled 140
forwarding 140
learning 140
listening 140
protected vlan 137
PVST+
description 148
mode 139
rapid root failover 142
rules and restrictions 159
StpdID 139, 160
terms 136
STPD modes 138
stub area, OSPF 190
switch, monitoring 91
syntax, understanding 21
system contact, SNMP 40
system location, SNMP 40
system name, SNMP 40

T

TACACS+
and RADIUS 48, 130, 132
description 48, 132
servers, specifying 132
tagging, VLAN 64
technical support 241
Telnet
connecting to another host 35
disconnecting a session 38
maximum sessions 35
opening a session 35
using 35
Terminal Access Controller Access Control System Plus. *See* TACACS+
TFTP
connecting to another host 38
enabling the server 39
server 227
using 231
traceroute 30
`traceroute` command 30
traffic groupings 85
transmit errors 94
triggered updates 187
trunks 65

U

upgrading the image 227
uploading the configuration 231
user name 43
users
access levels 27
authenticating 48, 129
creating 29
default 28
viewing 29

V

video applications, and QoS 83
viewing accounts 29
Virtual LANs. *See* VLANs
virtual link, OSPF 191
virtual router, VRRP 166
VLAN tagging 64
VLANs
and STP 137
assigning a tag 65
benefits 61
configuration examples 71
configuring 71
default 70
description 18
disabling route advertising 187
displaying settings 72
IP fragmentation 57
mgmt 35
mixing port-based and tagged 67
names 70
port-based 62
protocol filters 68
protocol-based 67
renaming 70
routing 179
tagged 64
trunks 65
types 62
voice applications, QoS 82
VRRP
advertisement interval 168, 171
and Spanning Tree 169
backup router 166
configuration parameters (table) 171
default gateway 165
description 165
electing the master 168

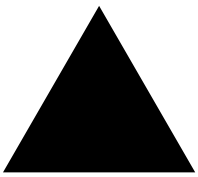
examples 172
interfaces 166
IP address 166, 171
IP address owner 166
MAC address 166
master
determining 166
master down interval 168, 171
master router 166
multicast address 168
operation 169
preempt mode 171
priority 166, 168, 171
redundancy 170
route table tracking 166
skew time 168, 171
tracking, description 166
virtual router 166
virtual router identifier (VRID) 166, 171
virtual router MAC address 166, 168, 169
VRRP router 166

W

web browsing applications, and QoS 83

X

xmodem 227
XOS shell, maximum sessions 34



Index of Commands

C

clear counters 110
clear log counters 110
clear session 25, 38
clear slot 54, 236
configure account 25
configure banner 26
configure bgp add aggregate-address 211
configure bgp add network 215
configure bgp neighbor dampening 213
configure bgp neighbor peer-group 212
configure bgp peer-group no dampening 213
configure bootprelay add 182
configure bootprelay delete 182
configure cli max-session 34
configure diffserv examination code-point 89
configure dns-client add 29
configure dns-client default-domain 29
configure dot1p type 87
configure dot1q ethertype 239
configure fdb agingtime 78
configure igmp snooping add static group 219
configure igmp snooping add static router 220
configure igmp snooping delete static group 220
configure igmp snooping delete static router 220
configure igmp snooping filter 220
configure iparp add proxy 178
configure iproute add default 35, 37, 180
configure iproute priority 179
configure jumbo-frame size 56
configure log filter 105
configure log filter events match 108
configure log target filter 103, 107
configure log target format 108
configure log target match 106
configure node offline 98
configure node online 98
configure node priority 98
configure osfp area nssa 191
configure osfp area stub 190
configure osfp ase-limit 189
configure ospf area timer 197
configure ospf timer 197
configure ospf virtual-link timer 197
configure ospf vlan area 190
configure ospf vlan timer 195, 198
configure pim add vlan 220
configure ports auto off 26, 55
configure ports auto on 55
configure ports qosprofile 89
configure protocol add 69
configure radius server client-ip 130
configure radius shared-secret 130, 131
configure radius timeout 130
configure radius-accounting 131
configure radius-accounting timeout 131
configure slot module 26, 53, 236
configure snmp add community 40
configure snmp add trapreceiver community 40
configure snmp delete trapreceiver 40
configure snmpv3 add access 43
configure snmpv3 add filter subtree type 47
configure snmpv3 add filter-profile param 47
configure snmpv3 add group user 44
configure snmpv3 add mib-view 45
configure snmpv3 add mib-view subtree 45
configure snmpv3 add notify tag 47
configure snmpv3 add target-addr param ipaddress 46
configure snmpv3 add target-params 42
configure snmpv3 add user 43
configure snmpv3 delete access 44
configure snmpv3 delete filter 47
configure snmpv3 delete filter-profile 47
configure snmpv3 delete group user 44
configure snmpv3 delete mib-view 45
configure snmpv3 delete notify 47
configure snmpv3 delete target-addr 46
configure snmpv3 delete target-params 46

configure snmpv3 delete user 43
configure snmpv3 engine-boots 43
configure snmpv3 engine-id 42
configure snmpv3 target-params user mp-model 46
configure sntp-client 50
configure sntp-client update-interval 50
configure stpd add vlan 140, 159
configure stpd default-encapsulation 139
configure stpd delete vlan 141
configure stpd mode 138
configure stpd port link-type 150, 151
configure stpd ports mode 139
configure stpd tag 159
configure sys-health-check interval 96
configure telnet port 35
configure tftp port 39
configure time 26
configure timezone 26, 49
configure vlan add ports 140
configure vlan ipaddress 37, 180
configure vlan ipadress 26
configure vlan name 70
create account 26, 29
create bgp neighbor peer-group 212
create bgp peer-group 211
create log filter 105
create ospf area 190
create protocol 68
create stpd 137, 159
create vlan 26

D

delete account 26, 29
delete bgp peer-group 211
delete stpd 137
delete vlan 26
disable bgp export 215
disable bootp vlan 26, 36
disable clipaging 26
disable dhcp vlan 36
disable edp ports 60
disable idletimeouts 26
disable learning ports 77
disable log debug-mode 240
disable log target 101
disable ospf capability opaque-lsa 189
disable ospf export 177, 195
disable ports 26, 54
disable radius 130
disable radius-accounting 131
disable rip export 195
disable rip exportstatic 177
disable sharing 58
disable snmp access 39
disable stpd rapid-root-failover 142
disable sys-health-check 96, 240
disable telnet 26, 39
disable tftp 39
disable udp-echo-server 183
download bootrom 29
download configuration 29
download image 29, 227

E

enable bgp aggregation 211
enable bgp export 215
enable bgp neighbor remove-private-as-numbers 214
enable bootp vlan 26, 36
enable bootprelay 182
enable clipaging 26
enable dhcp vlan 36
enable diffserv examination ports 88
enable edp ports 60
enable idletimeouts 26
enable ipforwarding 180
enable ipmcforwarding 220
enable jumbo-frame ports 56
enable log debug-mode 111, 240
enable log target 101
enable log target console 109
enable log target session 109
enable ospf 180
enable ospf capability opaque-lsa 189
enable ospf export 195
enable ospf export static 177
enable pim 220
enable ports 54
enable radius 130
enable radius-accounting 131
enable rip 180
enable rip export 195
enable rip exportstatic 177
enable sharing grouping 58
enable snmp access 39
enable sntp-client 50
enable stpd 159
enable stpd auto-bind 141
enable stpd rapid-root-failver 142
enable sys-health-check 96, 240
enable tftp 39
enable udp-echo-server 183

F

failover 98

H

history 25, 26

L

logout 38
ls 231

M

mtrace 220
mv 231

N

nslookup 29

P

ping 27, 29, 30

Q

quit 38

R

reboot 229
rm 232
run diagnostics 92

S

save configuration 230
show accounts 29
show banner 26
show bgp neighbor 213
show bgp peer-group 213
show checkpoint-data 100
show dhcp-client state 36
show edp 60
show fans 236
show fdb 78
show igmp snooping filter 220
show igmp snooping static group 220
show iparp 180
show ipconfig 180, 182
show iproute 180
show log 109
show log components 103
show log configuration filter 105
show log configuration target 102
show log counters 110
show log events 104
show management 40
show odometer 240
show ospf 195, 200
show ospf area 201
show ospf interfaces 201
show ospf lsdb 201
show ospf lsdb area lstype 201
show ports info 89, 90
show ports qosmonitor 90
show ports rxerrors 94
show ports sharing 59
show ports stats 93
show ports txerrors 94
show powersupplies 236
show protocol 73
show qosprofile 89, 90
show session 38
show slot 54
show snmpv3 access 43
show snmpv3 filter 47
show snmpv3 filter-profile 47
show snmpv3 group 44
show snmpv3 mib-view 45
show snmpv3 notify 47
show snmpv3 target-addr 46
show snmpv3 target-params 46
show snmpv3 user 43
show sntp client 50
show stpd 142, 163
show stpd ports 151, 164
show switch 49, 50, 228, 240
show temperature 95
show version 228
show vlan 72, 239
show vlan stpd 164

T

telnet 29, 35, 38
tftp 231
traceroute 29, 30

U

unconfigure switch 27, 230
upload log 109
use configuration 230
use image 228